15.002/H/02

TUGAS AKHIR

(KP 1701)

PENGGUNAAN TEKNIK TOPOLOGICAL PATTERN (GRAPH) MATCHING DALAM PEMILIHAN DESAIN TATA LETAK GALANGAN KAPAL

RSPe
623.83
Pra
P-1
2001

OLEH :

IRAWAN PRASETYO

NRP. 4198 100 511

JURUSAN TEKNIK PERKAPALAN
FAKULTAS TEKNOLOGI KELAUTAN
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
S U R A B A Y A

2001

PERPUSTAKAAN
I T S
Tgl. Terima 2-1-2002
Terima Dari H
No. Agenda Prp 24782

PENGGUNAAN TEKNIK TOPOLOGICAL PATTERN (GRAPH) MATCHING DALAM PEMILIHAN DESAIN TATA LETAK GALANGAN KAPAL

TUGAS AKHIR

Telah Direvisi Sesuai dengan Hasil Sidang Ujian Tugas Akhir

Pada

Jurusan Teknik Perkapalan

Fakultas Teknologi Kelautan

Institut Teknologi Sepuluh Nopember

Surabaya

Surabaya, Agustus 2001

Mengetahui / Menyetujui

Dosen Pembimbing

Ir. Djauhar Manfaat, M.Sc., Ph.D.

NIP. 131 651 444

PENGGUNAAN TEKNIK TOPOLOGICAL PATTERN (GRAPH) MATCHING DALAM PEMILIHAN DESAIN TATA LETAK GALANGAN KAPAL

TUGAS AKHIR

Diajukan Guna Memenuhi Sebagian Persyaratan Untuk Memperoleh Gelar

Sarjana Teknik Perkapalan

Pada

Jurusan Teknik Perkapalan
Fakultas Teknologi Kelautan
Institut Teknologi Sepuluh Nopember
Surabaya

Surabaya, Juli 2001

Mengetahui / Menyetujui

Dosen Pembimbing

Ir. Djauhar Manfaat, M.Sc., Ph.D.

Nip. 131 651 444

INSTITUT TEKNOLOGI SEPULUH NOPEMBER (ITS)

ABSTRAK

FAKULTAS TEKNOLOGI KELAUTAN
JURUSAN TEKNIK PERKAPALAN

Sarjana Teknik (S1)

PENGGUNAAN TEKNIK TOPOLOGICAL PATTERN (GRAPH) MATCHING DALAM PEMILIHAN DESAIN TATALETAK GALANGAN KAPAL

Oleh : Irawan Prasetyo
Pembimbing : Ir. Djauhar Manfaat, M.Sc., Ph.D.

Salah satu tahapan dari desain suatu sistem produksi pada galangan kapal adalah perencanaan tata letak galangan kapal yang merupakan fase awal dan terpenting untuk proses selanjutnya yang hasil desainnya akan mempengaruhi efisiensi, pembentukan laba dan kelangsungan dari galangan kapal tersebut.

Adanya permasalahan waktu penyelesaian perencanaan dari sebuah tata letak merupakan pertanyaan yang harus dapat dijawab oleh seorang desainer untuk dapat mengembangkan ide atau gagasannya sehingga diperoleh penyelesaiannya. Salah satu cara adalah dengan memanfaatkan desain yang sudah ada sebagai referensi dalam menyelesaikan sebuah desain baru.

Tujuan dalam penulisan ini adalah mengurangi waktu dan proses perencanaan dengan menggunakan salah satu teknik dalam Case Base Reasoning (CBR) yaitu teknik Topological Pattern (Graph) Matching yang diaplikasikan dalam sebuah sistem yang dikembangkan dan dinamakan AISY (Artificial Intelligence for Design Retrieval of Shipyard Layouts).

Dengan pendekatan CBR akan mewujudkan keinginan dari desainer tersebut untuk dapat mengurangi beban kerjanya. Pengurangan itu adalah waktu yang diperlukan dalam proses sehingga menjadi lebih singkat karena desainer tidak perlu melakukan perencanaan dari awal. Dengan itu, desainer dapat lebih menitikberatkan pekerjaannya untuk meningkatkan mutu desain.

Melalui penerapan teknik Topological Pattern (Graph) Matching akan didapatkan relevansi terbaik antara desain-desain tata letak lama dengan desain baru sehingga diperoleh angka kesamaan tetinggi. Dengan dukungan komputer sebagai alat bantu dalam mengimplementasikan teknik di atas akan dipresentasikan sistem AISY yang akan mewujudkan pemanfaatan desain-desain lama menjadi solusi dari proses desain baru.

Dari hasil yang diperoleh dalam pengujian terhadap sistem AISY dengan objek layout galangan kapal menunjukkan bahwa sistem tersebut mempunyai kemampuan dan konsistensi dalam pelaksanaan proses pattern matching antara template dengan data.

SEPULUH NOPEMBER INSTITUTE OF TECHNOLOGY (ITS)

ABSTRACT

FACULTY OF MARINE TECHNOLOGY
DEPARTMENT OF NAVAL ARCHITECTURE AND SHIPBUILDING

Degree in Engineering (S1)

IMPLEMENTING TOPOLOGICAL PATTERN (GRAPH) MATCHING TECHNIQUE FOR DESIGN RETRIEVAL OF SHIPYARD LAYOUTS

By : Irawan Prasetyo
Supervisor : Ir. Djauhar Manfaat, M.Sc., Ph.D.

One of the parts of design process on the shipyard was spatial arrangement. This part is the beginning of the stages of design. The stage will indicate the efficiency of the work, profit and production process.

Amount of time is needed of design in planning of shipyard is challenge for the designer in developing his idea to achieve the final solution. One of the techniques to create a new design is to compare with the past designs which is used as a reference.

The objective of this study was define a new approach to reducing time and design process for spatial arrangement by using the Topological Pattern Matching (Graph) Matching through the Case Base Reasoning (CBR) approaching by implementing it in the AISY system.

CBR used in this study indicated that a shorter time was used to create a new space arrangement, since the designer does not necessarily to design the layout from the scratch. By using the CBR approach the designer could focus his work more on the quality of the new design.

The Topological Pattern (Graph) matching technique implemented through the AISY system gives the highest relevance to the past designs. This study showed that past designs help the designer to achieve the final solution.

In addition, the scope of this study is to design shipyard layouts. It revealed that the system is able and consistent to simulate data obtained from this study according to the pattern matching similar to manual work.

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, karena berkat rahmat dan karunianya maka penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul **"PENGGUNAAN TEKNIK *TOPOLOGICAL PATTERN (GRAPH) MATCHING* DALAM PEMILIHAN DESAIN TATA LETAK GALANGAN KAPAL".**

Tugas Akhir ini merupakan tugas dan kewajiban guna melengkapi sebagian syarat-syarat untuk mencapai gelar Sarjana Teknik Perkapalan.

Dalam penulisan Tugas Akhir ini penulis telah berusaha sebaik mungkin dalam menjabarkan permasalahan dan penerapan penyelesaian atas permasalahan tersebut sesuai dengan judul Tugas Akhir ini, namun penulis menyadari bahwa masih banyak kekurangan dan kelemahannya, sehingga dapatlah kiranya untuk dimaklumi.

Ucapan terima kasih penulis tujukan kepada :

1. Bapak Ir. Djauhar Manfaat M.Sc, Ph.D, selaku Ketua Jurusan Teknik Perkapalan dan Dosen Pembimbing Tugas Akhir yang telah banyak membimbing, mencurahkan ilmu, waktu dan tenaganya hingga penulis dapat menyelesaikan Tugas Akhir ini sesuai dengan waktunya.
2. Bapak Ir. I.K.A Pria Utama M.Sc, Ph.D, selaku Dosen Pembimbing Akademik.
3. Seluruh Staf Dosen dan Karyawan Jurusan Teknik Perkapalan atas bimbingan dan bantuannya selama penulis dalam pendidikan.

4. Ibu, Bapak serta adik-adikku tercinta yang telah memberikan dorongan moril dan materiil.
5. Rekan-rekan mahasiswa Teknik Perkapalan khususnya Lintas Jalur Angkatan ,'96,'97 dan '98 atas dukungannya selama ini. Semoga kompak selalu !!!!
6. Kartika Ari Nugraha dan Rina Pangastuti atas kerjasamanya yang baik dalam pembuatan program komputer.
7. Serta semua pihak yang telah membantu, yang tidak mungkin disebutkan satu persatu.

Akhirnya penulis berharap semoga Tugas Akhir ini dapat bermanfaat bagi yang memerlukannya.

Surabaya, Juli 2001

Penulis

DAFTAR ISI

Hal

JUDUL TUGAS AKHIR

LEMBAR PENGESAHAN

ABSTRAK i

KATA PENGANTAR iii

DAFTAR ISI v

DAFTAR GAMBAR vii

DAFTAR TABEL ix

BAB I PENDAHULUAN 1.1

1.1 Latar Belakang Masalah 1.1

1.2 Perumusan Masalah 1.3

1.3 Batasan Masalah 1.4

1.4 Tujuan dan Masalah 1.4

1.5 Metodologi dan Model Analisis 1.5

1.6 Sistematika Laporan 1.8

BAB II LANDASAN TEORI 2.1

2.1 Tata Letak (*Layout*) Galangan Kapal 2.1

2.2 Pemanfaatan Desain *Layout* Lama Untuk Proses Desain Baru 2.5

2.3 Komputer Sebagai Sarana Pendukung 2.9

2.4 *Case Based Reasoning (CBR)* Bidang Desain 2.11

2.4.1 Representasi Kasus Desain 2.13

2.4.2 Penarikan Kasus Desain 2.19

2.4.3 Adaptasi Kasus Desain 2.20

2.5 Pendekatan *Pattern Matching* 2.21

2.5.1 Klasifikasi *Pattern* 2.22

2.5.2 Metode *Matching* 2.27

2.5.3 Jenis Kesamaan 2.30

BAB III TEKNIK IMPLEMENTASI DARI *CBR* 3.1
3.1 Representasi Data 3.1
3.1.1 Representasi Ruang 3.1
3.1.2 Representasi *Layout* 3.5
3.2 *Topological Pattern (graph) Matching* 3.5
3.2.1 Aplikasi pada *layout* galangan kapal 3.9
BAB IV SISTEM AISY 4.1
4.1 Pemodelan Sistem 4.1
4.1.1 *Libraries* 4.2
4.1.2 *Layout Viewer* 4.3
4.1.3 *Pattern Matching* 4.3
4.1.4 Sistem *Interface* 4.3
BAB V ANALISA HASIL 5.1
BAB VI PENUTUP 6.1
6.1 Kesimpulan 6.1
6.2 Saran 6.2
DAFTAR PUSTAKA
LAMPIRAN

DAFTAR GAMBAR

Hal

Gambar 1.1 Metodologi penelitian..... 1.7

Gambar 2.1 *Layout* tipe dasar I dan T..... 2.2

Gambar 2.2 *Layout* tipe dasar L..... 2.2

Gambar 2.3 *Layout* tipe dasar U..... 2.2

Gambar 2.4 *Layout* tipe dasar Z..... 2.2

Gambar 2.5 Representasi *layout* galangan kapal tipe dasar I. 2.7

Gambar 2.6 Diagram tahapan proses produksi..... 2.8

Gambar 2.7 Representasi *layout* galangan berdasarkan urutan proses produksi..... 2.9

Gambar 2.8 Model *case base reasoning* sederhana dari Maher, dkk. 1995..... 2.13

Gambar 2.9 Empat bagian dasar dari *patterns* (Manfaat dkk., 1996)..... 2.22

Gambar 2.10 Ilustrasi dari contoh gambar pada kelas *geometric pattern* (Manfaat dkk., 1996)..... 2.24

Gambar 2.11 Macam-macam contoh dari *spatial pattern* yang mempunyai bentuk *pixel* 8 x 8 (Manfaat dkk., 1996)..... 2.25

Gambar 2.12 Contoh representasi *pixel* pada *spatial relationship pattern* (Manfaat dkk., 1996)..... 2.26

Gambar 3.1 Representasi ruang segi empat dan poligon (Manfaat, 1998).. 3.3

Gambar 3.2 Representasi sebuah *graph* dengan beberapa properti dan

Hubungan (Manfaat dkk., 1998) 3.7

Gambar 3.3 Contoh dua buah *layout* pada *catering deck* (Manfaat dkk., 1998) 3.7

Gambar 3.4 Representasi dalam bentuk *graph* (Manfaat dkk., 1998) 3.7

Gambar 3.5 Gabungan *graph* hasil *matching* dari dua *layout* di atas (Manfaat dkk., 1998) 3.8

Gambar 3.6 Tiga macam kumpulan *layout* galangan kapal dalam *case base* 3.11

Gambar 3.7 *Layout template* 3.12

Gambar 4.1 Pemodelan sistem AISY 4.2

Gambar 4.2 Menu utama sistem AISY 4.4

Gambar 4.3 Window untuk *Shipyard-Layout* 4.5

Gambar 4.4 Window untuk Data *Shipyard-Layout* 4.5

Gambar 4.5 Window untuk *Layout-Templates* 4.5

Gambar 4.6 Window untuk *Shipyard Layout-2* 4.6

Gambar 4.7 Window untuk *Layout Templates-1* 4.6

Gambar 4.8 *Most similarity antara Layout Template-2* dengan *Ship Yard Layouts* 4.7

Gambar 4.9 Kesamaan hubungan ruangan yang didapat 4.7

Gambar 4.10 Hasil matching antara *Layout Template-2* dengan semua data *layout* dalam *case base* 4.8

DAFTAR TABEL

Hal

Tabel 3.1 Hasil kesesuaian antara *layout template* pada gambar 3.7 dengan kumpulan *layout* galangan kapal dalam *case base* pada gambar 3.6 ..... 3.12

Tabel 5.1 *Matching Result* antara *Layout Template-1* dengan data *Layout* galangan kapal *(Shipyard Layouts)* 5.2

Tabel 5.2 *Matching Result* antara *Layout Template-2* dengan data *layout* galangan kapal *(Shipyard Layouts)* 5.3

Tabel 5.3 *Matching Result* antara *Layout Template-3* dengan data *layout* galangan kapal *(Shipyard Layouts)* 5.4

BAB I

PENDAHULUAN

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG MASALAH

Selama ini penentuan/perencanaan tata letak pabrik, dalam hal ini adalah galangan kapal dilakukan secara konvensional dengan mempertimbangkan peraturan-peraturan yang ada dan juga berdasarkan pengalaman atau data pembanding yang ada sehingga hasil perencanaan sangat tergantung dari selera perencana dan data pembanding yang ada. Perkembangan dari teori-teori perencanaan tata letak dan teori pemindahan bahan yang umumnya diterapkan dalam perencanaan tata letak pabrik dan fasilitas-fasilitas yang ada di dalamnya (Apple, 1990). Dalam teori ini dilakukan suatu rancangan susunan fasilitas-fasilitas produksi yang ada dengan mempertimbangkan masukan-masukan dan proses yang dibutuhkan sehingga akan dicapai *output* yang diinginkan. Maka dengan teori ini akan didapatkan tata letak fasilitas pabrik yang baik dan waktu pemindahan bahan yang minimum sehingga dapat meningkatkan hasil produksi.

Permasalahan-permasalahan utama yang dialami desainer bila menggunakan cara seperti tersebut di atas adalah :

- Memerlukan waktu yang lama.

  Disini desainer memilih desain yang sudah ada dan disesuaikan dengan keinginan dari *owner* secara manual.

- Hasil yang didapat kurang memuaskan karena adanya kekurangtelitian dari desainer dalam memilih desain lama yang sesuai dengan permintaan pemesan.

Permasalahan-permasalahan tersebut di atas dapat dikurangi jika pemanfaatan atas desain-desain lama dilakukan. Dengan menggunakan pendekatan *case based reasoning* melalui teknik *topological patern* akan dihasilkan desain tata letak yang lebih baik dalam waktu yang relatif singkat dengan meminimumkan pergerakan material sehingga dengan demikian dapat menekan *cost*.

Menurut Manfaat (1992), pemanfaatan desain-desain lama untuk proses desain baru mempunyai beberapa keuntungan, antara lain :

1. Dapat mempercepat proses desain.

   Disini proses desain tidak dimulai dari awal karena desainer menggunakan desain-desain lama yang sudah ada, sehingga dapat mempercepat waktu perancangan.

2. Mengurangi beban kerja desainer, karena desainer tidak perlu merancang desain dari awal (*from scratch*).

3. Hasil desain yang dihasilkan lebih mendekati permintaan pemesan (*owner*).

   Hal tersebut disebabkan karena desainer mengumpulkan desain-desain lama yang mempunyai kesamaan-kesamaan tata letak dengan desain dari *owner*. Dari desain-desain tersebut desainer akan memilih desain dengan angka kesamaan tertinggi.

CBR merupakan pendekatan yang dapat mewujudkan keuntungan-keuntungan di atas. Maher dkk. (1995) memperkenalkan sebuah model CBR (*in design*) mulai dari proses penyimpanan sampai dengan proses adaptasi (lihat Bab II). Awalnya data-data *layout* galangan kapal dikumpulkan dan disimpan dalam

case base karena seorang desainer tentunya membutuhkan penyimpanan data secara lengkap, aman dan akurat. Atas permintaan *owner* dalam mendesain tata letak galangan kapal baru, desainer tersebut cukup hanya dengan mencari data-data lama dari dalam *case base*. Dengan proses penarikan data *layout* galangan kapal dari dalam *case base* secara cepat, akurat dan rapi maka akan memudahkan desainer untuk mencari kesamaan data tata letak yang sesuai dengan permintaan *owner*. Selanjutnya desain dengan nilai kesamaan tertinggilah yang akan dipilih untuk kemudian diadaptasi menjadi desain baru.

Teknik yang digunakan dalam penyajian data tata letak galangan kapal dari dalam *case base* menggunakan teknik *topological pattern (graph) matching*, yaitu merepresentasikan data-data tata letak tersebut dalam bentuk graph dengan mencari hubungan korespondensi (*clique*) terbanyak dari masing-masing data tersebut. Untuk memudahkan pengerjaan tahap - tahap perancangan dengan *case base reasoning*, digunakan program komputer *LISP* yang akan membantu dalam proses penarikan dan penyajian hasil.

1.2 PERUMUSAN MASALAH

Berdasarkan uraian latar belakang di atas, penulis dapat menguraikan permasalahan sebagai berikut :

1. Bagaimana merepresentasikan tata lelak galangan kapal yang sudah ada ke dalam *case base*.
2. Bagaimana merepresentasikan desain tata letak ke dalam bentuk *graph*.

3. Bagaimana menentukan kesamaan desain permintaan owner (*templates*) dengan desain-desain yang ada dalam *case base* sehingga diperoleh jumlah *clique* terbayak.
4. Bagaimana mengimplementasikan teknik *topological pattern (graph) matching* pada program komputer.

1.3 BATASAN MASALAH

Batasan masalah yang digunakan dalam tugas akhir ini adalah :

- Pendekatan CBR yang digunakan dibatasi sampai dengan proses penarikan (*recall*) saja.
- Data *layout* galangan kapal dibatasi sampai dengan kapasitas 50000 ton.
- Data *layout* galangan kapal yang dibutuhkan minimal 10 buah dan hanya merupakan objek saja, yang untuk selanjutnya akan dilakukan analisa.
- Luas area dari sebuah ruang atau fasilitas pada galangan kapal tidak dianalisa tetapi akan dicantumkan sebagai informasi dari desainer.

1.4 TUJUAN DAN MANFAAT

Tujuan utama dari penulisan tugas akhir ini adalah untuk memperbaiki proses desain tata letak galangan dengan teknik *topological pattern (graph) matching*.

Tujuan spesifik yang dapat mendukung tujuan utama di atas yaitu

a. Untuk merepresentasikan desain tata letak galangan kapal dalam bentuk *graph*.

b. Untuk mengembangkan teknik *topological pattern (graph) matching* tata letak galangan kapal.

c. Teknik pengukuran kesamaan/hubungan korespondensi (*clique*) tata letak galangan kapal.

d. Mengimplementasikan teknik-teknik di atas ke dalam prototype komputer.

Sedangkan manfaat dari penulisan tugas akhir ini adalah :

Bagi desainer :

a. Mengurangi tugas desainer, karena desainer tidak merancang desain dari awal.

b. Mempercepat waktu perancangan desain tata letak.

Bagi *owner* / pemilik galangan :

a. Desain yang dihasilkan lebih mendekati keinginan *owner*.

b. Meningkatkan sistem/proses produksi dengan penempatan peralatan yang lebih baik dan pergerakan material minimal sehingga diperoleh produk yang maksimal.

c. Mengurangi biaya produksi (*cost*) sehingga keuntungan galangan lebih besar.

1.5 METODOLOGI DAN MODEL ANALISIS

Metodologi yang digunakan dalam penulisan tugas akhir ini disertai dengan model analisisnya adalah sebagai berikut :

1. Studi Literatur

Dilakukan studi literatur tentang ruang-ruang/fasilitas-fasilitas pada galangan kapal, aspek-aspek desain yang harus diperhatikan, pendekatan *CBR* dengan teknik *topological pattern (graph) matching* dan *LISP* sebagai perangkat lunak

untuk mengimplementasikan teknik di atas yang akan dipakai yang digunakan sebagai dasar pemilihan tata letak galangan kapal mulai dari perumusan latar belakang sampai penulisan (dokumentasi) tugas akhir ini.

2. Perumusan latar belakang

   Pada tahap ini dilakukan perumusan latar belakang yang berkaitan dengan permasalahan yang ingin dipecahkan.

3. Hipotesis awal

   Tahap ini merupakan dugaan awal terhadap hasil yang ingin dicapai.

4. Perumusan masalah

   Merupakan tahapan merumuskan masalah-masalah yang timbul waktu mengerjakan tugas akhir ini.

5. Pemecahan masalah

   Pada tahap ini dilakukan pencarian solusi terhadap permasalahan-permasalahan yang telah dirumuskan sebelumnya.

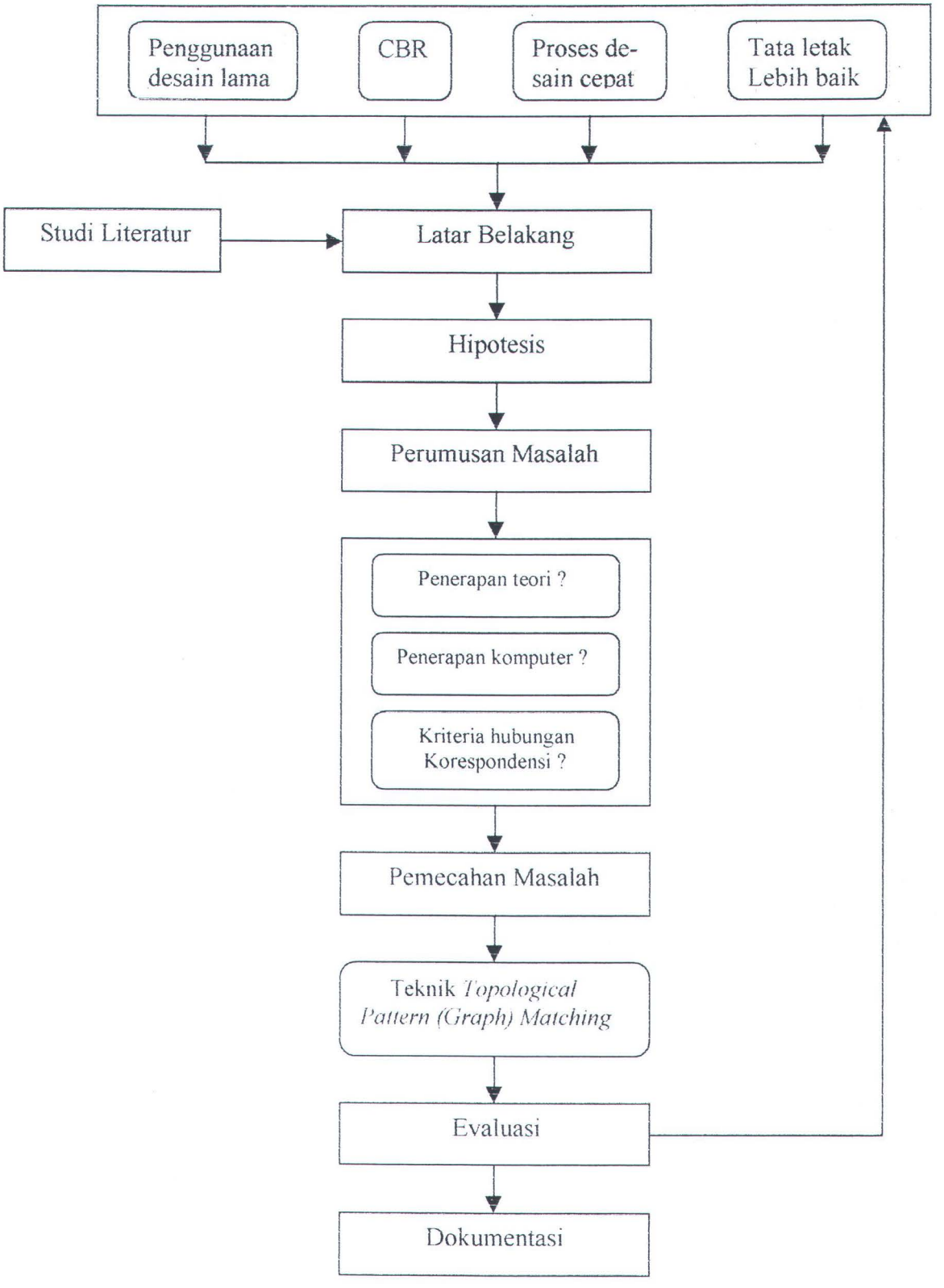
6. Evaluasi

   Hasil yang dicapai pada tahap pemecahan masalah di atas dievaluasi.

7. Dokumentasi

   Merupakan tahap terakhir dalam tugas akhir ini yaitu tahap penulisan tugas akhir terhadap proses penelitian yang telah dilakukan di atas.

Sedangkan *flow chart* metodologi penelitian dapat dilihat pada halaman berikut :



Gambar 1.1 Metodologi Penelitian

1.6 SISTEMATIKA LAPORAN

Sistematika pembahasan dalam penulisan tugas akhir ini dibagi menjadi enam bab, yang berisi pokok-pokok uraian sebagai berikut :

BAB I : Merupakan bab pendahuluan yang mencakup latar belakang permasalahan, perumusan masalah, batasan masalah, tujuan dan manfaat, metodologi penelitian serta sistematika laporan yang digunakan dalam penulisan tugas akhir ini.

BAB II : Mengetengahkan pengertian-pengertian dari tata-letak *(layout)* galangan kapal, pemanfaatan desain *layout* lama untuk proses desain baru, komputer sebagai sarana pendukung dan *case based reasoning (CBR)* sebagai bagian dari *artificial intelligence* yang merupakan pemecahan masalah dari latar belakang di atas.

BAB III : Isi bab ini merupakan implementasi dari pendekatan *CBR* yang meliputi representasi data, teknik *topological pattern (graph) matching* sebagai bagian utama dari pendekatan pattern matching dan sebagai teknik penunjang dalam proses *recall* disertai dengan aplikasinya pada *layout* galangan kapal.

BAB IV : Bab ini berisi tentang sistem AISY yang akan menjalankan pemanfaatan desain lama untuk proses desain baru *(design reuse)* sampai dengan tahap penarikan desain *(recall)*. Komponen-komponen utama dari sistem ini akan dijelaskan disertai dengan tampilan windows yang menarik.

BAB V : Bab ini merupakan penjelasan analisa hasil berkaitan dengan hasil proses matching yang telah dilakukan.

BAB VI : Mengemukakan beberapa kesimpulan yang dapat ditarik dari hasil pembahasan yang penulis lakukan serta beberapa saran yang dapat disampaikan sebagai sumbangan pemikiran.

BAB II

LANDASAN TEORI

ITS

BAB II

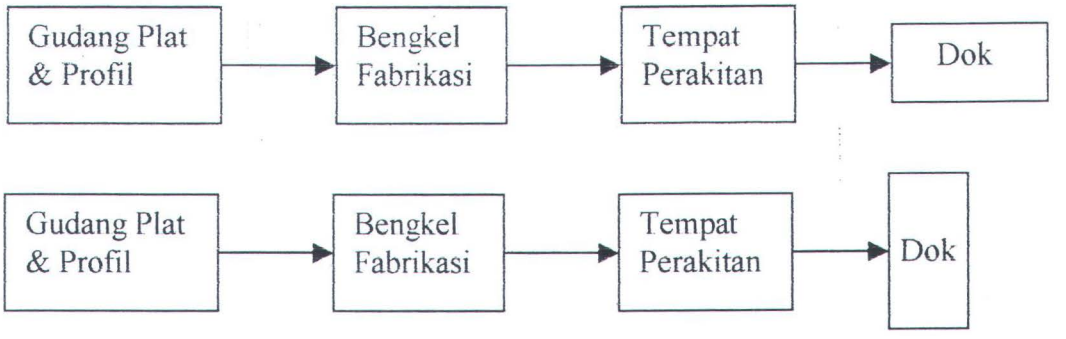
LANDASAN TEORI

Dalam bab ini akan diuraikan teori-teori pendukung sebagai penunjang proses desain baru dimulai dengan tata letak galangan kapal dimana segi-segi dalam perencanaan tata letak galangan yang meliputi tipe/bentuk, cara pengaturan *layout* dan faktor-faktor penyusun *layout* serta tahapan dalam penyusunan *layout* sebagai dasar pengetahuan yang harus dimiliki oleh seorang desainer, kemudian bagaimana seorang desainer memanfaatkan desain *layout* lama untuk proses desain baru (*design reuse*), komputer sebagai sarana pendukung dan pendekatan *Case Based Reasoning (CBR)* sebagai solusi pemecahannya.

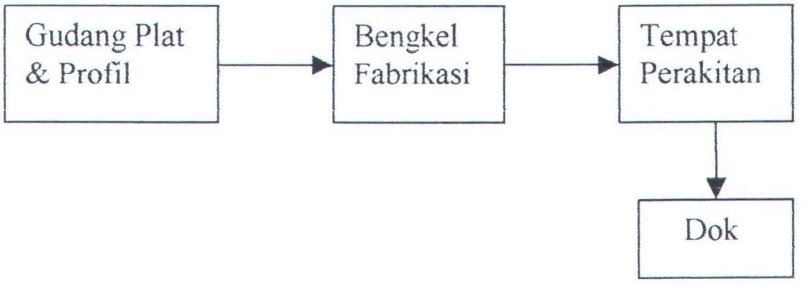
2.1 TATA LETAK (*LAYOUT*) GALANGAN KAPAL

Tata letak dapat didefinisikan sebagai pengaturan/peletakan *subspace-subspace* dalam suatu *space* (Manfaat, 1992). Atau dalam tata letak galangan kapal, tata letak dapat didefinisikan sebagai pengaturan fasilitas-fasilitas produksi dalam suatu lahan/area. Sedangkan menurut Andjar Soeharto dan Soejitno (1996), tata letak (*layout*) galangan kapal merupakan sederetan letak fasilitas produksi. Dalam pengaturannya didasarkan atas proses produksi dengan mempertimbangkan kondisi lokasi yang ada dan rencana pengembangan di masa mendatang. Tata letak (*layout*) galangan kapal dapat diklasifikasikan menjadi empat tipe dasar, yaitu layout tipe I atau T, tipe L, tipe U dan tipe Z (Andjar Soeharto dan Soejitno, 1996) seperti yang diuraikan pada gambar di bawah ini :

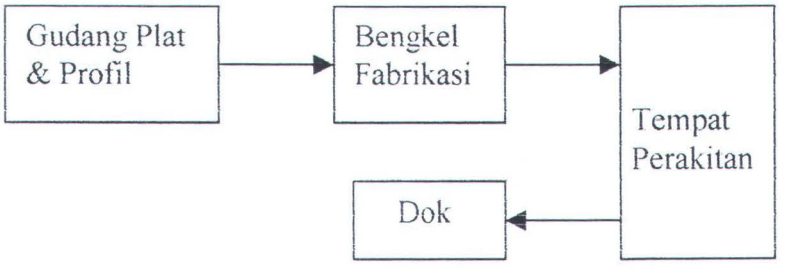
Sebutkan nama gambar.
(96.2.1, 2.2, 2.3)



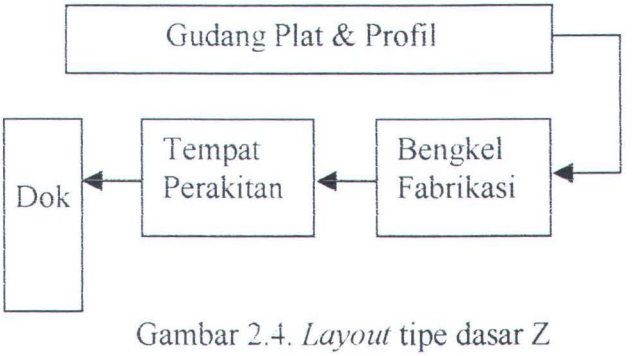
Gambar 2.1. *Layout* tipe dasar I dan T



Gambar 2.2. *Layout* tipe dasar L



Gambar 2.3. *Layout* tipe dasar U



Gambar 2.4. *Layout* tipe dasar Z

Pada dasarnya penyusunan layout galangan harus disesuaikan dengan :

- Lokasi galangan
- Ukuran dan tipe kapal yang akan dibangun atau direparasi
- Metode pembangunan kapal
- Skala produksi

Perencanaan *layout* adalah merupakan tahapan awal yang termasuk dalam desain dari suatu sistem produksi. *Layout* galangan kapal yang dihasilkan nantinya akan mempengaruhi efisiensi dari galangan kapal, pembentukan laba galangan kapal dan kelangsungan galangan kapal.

Tujuan dari penataan sarana produksi adalah :

- Mengurangi jarak *material handling*
- Tidak terganggunya frekwensi produksi
- Mempermudah perawatan sarana produksi
- Menekan investasi dan ongkos produksi
- Mempertinggi keselamatan kerja
- Mempertinggi efisiensi produksi
- Mempertinggi mutu hasil produksi
- Memudahkan pengawasan

Jadi tata letak (*layout*) galangan kapal yang baik dapat diartikan sebagai penyusunan yang teratur dan efisien semua fasilitas-fasilitas pabrik dan pekerja yang ada dalam galangan serta pergerakan material minimal sehingga akan menurunkan biaya produksi (*cost*).

Cara-cara pengaturan *layout* adalah :

- Atas dasar proses (*process lay out/functional layout*)
- Atas dasar *arus /flow* (*product lay out/flow or line layout*)
- Kombinasi keduanya

Process layout dapat diartikan bahwa semua mesin-mesin dan peralatan yang sama ditempatkan/dikelompokkan dalam suatu tempat yang sama untuk memproduksi barang yang tidak sama dengan jumlah terbatas. Mesin yang digunakan adalah tipe *general purpose machine*, contohnya adalah bengkel reparasi. Sedangkan pada *product layout*, mesin-mesin disusun berdasarkan urutan proses produksi, contohnya adalah galangan khusus bangunan baru. Untuk kombinasi keduanya dapat dilihat pada galangan untuk bangunan baru dan reparasi.

Faktor-faktor untuk menyusun *layout* galangan kapal :

- Produk yang dihasilkan
- Urutan produksi yang sesuai dengan proses produksi
- Kebutuhan ruangan yang cukup luas
- Peralatan/mesin-mesin itu sendiri
- *Maintenance* dan *replacement*
- Keseimbangan kapasitas (*balance kapacity*)
- Minimum pergerakan material
- Aliran material
- Tempat kerja karyawan
- *Service area*, yaitu tempat perakitan atau *assembly yard*

- *Waiting area*, yaitu tempat menyimpan material untuk menunggu proses selanjutnya
- *Plane climate*, yaitu pengaturan udara dan suhu dalam pabrik
- *Flexibility*, yaitu fasilitas produksi dapat disesuikan pada kapasitas galangan yang lebih tinggi

Beberapa tahapan dalam menyusun *layout*, yaitu :

- *Plant inventory* berupa daftar mesin, dimensi mesin dan gambar mesin (skala).
- *Group outline*, yaitu dalam perancangan harus diperhatikan macam-macam mesin secara kelompok, yaitu mesin-mesin yang sama jenis dan ukurannya.
- Alat-alat pembantu seperti *material handling* dan alat lainnya.
- Metode *investigation*, dimana ruang gerak hasil produksi, alat transport, gang-gang harus cukup agar tidak saling mengganggu.
- Daerah mesin, adalah ruang untuk *maintenance*.
- Machine block plan, yaitu pengaturan mesin sesuai dengan proses poduksi.
- *Shop floor layout*, yaitu penentuan *layout machine block* yang ditinjau dari segi *flow of production*, pembagian gang, dimensi *machine shop*, kedudukan penghalang dan penempatan gudang.

2.2 PEMANFAATAN DESAIN *LAYOUT* LAMA UNTUK PROSES DESAIN BARU

Pengetahuan tentang desain lama akan mengingatkan desainer pada kasus desain lama tertentu. Pengetahuan tersebut merepresentasikan atribut desain, penyelesaian, proses penyelesaian dan pengalaman desainer. Atribut desain

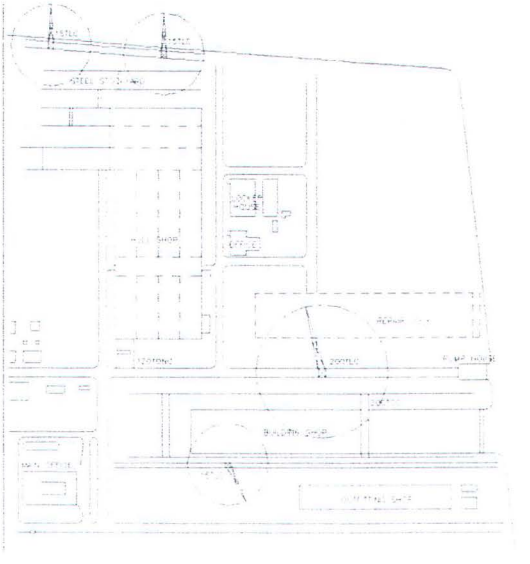
mempunyai bentuk meliputi ukuran/dimensi, luasan, perbandingan luasan, bentuk, fungsi dan hubungan kedekatan antar ruang. Atribut desain tersebut akan dicari kesesuaiannya dengan persyaratan desain baru dan dapat dipergunakan kembali untuk penyelesaian masalah desain baru sebagai pertimbangan (Andrews, 1985; Kolodner, 1993; Maher dkk., 1995).

Untuk penyelesaian desain lama, desainer dapat memperoleh dan menggunakan kembali desain lama tersebut untuk disesuaikan dengan syarat desain baru (Foz, 1973; Oxman and Oxman, 1992; Hua dkk., 1992; Maher dkk., 1995). Modifikasi atau adaptasi dari desain lama selalu menghasilkan pencapaian desain baru dan penyelesaian desain lama sering kali digunakan sebagai contoh (Oxman and Oxman, 1992). Tipe representasi dari penyelesaian desain-desain lama adalah bentuk gambar dua dimensi. Untuk menghasilkan kesamaan dengan desain lama, desainer mencari kesesuaian persyaratan desain baru dengan atribut desain lama. Dengan pengalaman yang dimilikinya, desainer dapat mengidentifikasi kelebihan dan kekurangan dari desain lama sehingga dapat dihasilkan desain baru.

Dalam pembuatan sebuah desain *layout* galangan baru, seorang desainer tentunya akan mengumpulkan semua arsip yang dimilikinya berupa desain-desain *layout* lama yang sudah pernah ia kerjakan sebelumnya atau bisa juga desain-desain *layout* yang sudah pernah dikerjakan orang lain yang kebetulan ia miliki. Desain-desain tersebut akan dipergunakan sebagai bahan pembanding untuk proses desain baru, tentunya juga dengan mempertimbangkan berbagai aspek

yang telah dijelaskan di atas (atribut desain) dan pada sub bab sebelumnya dalam penyusunan *layout* galangan kapal baru.

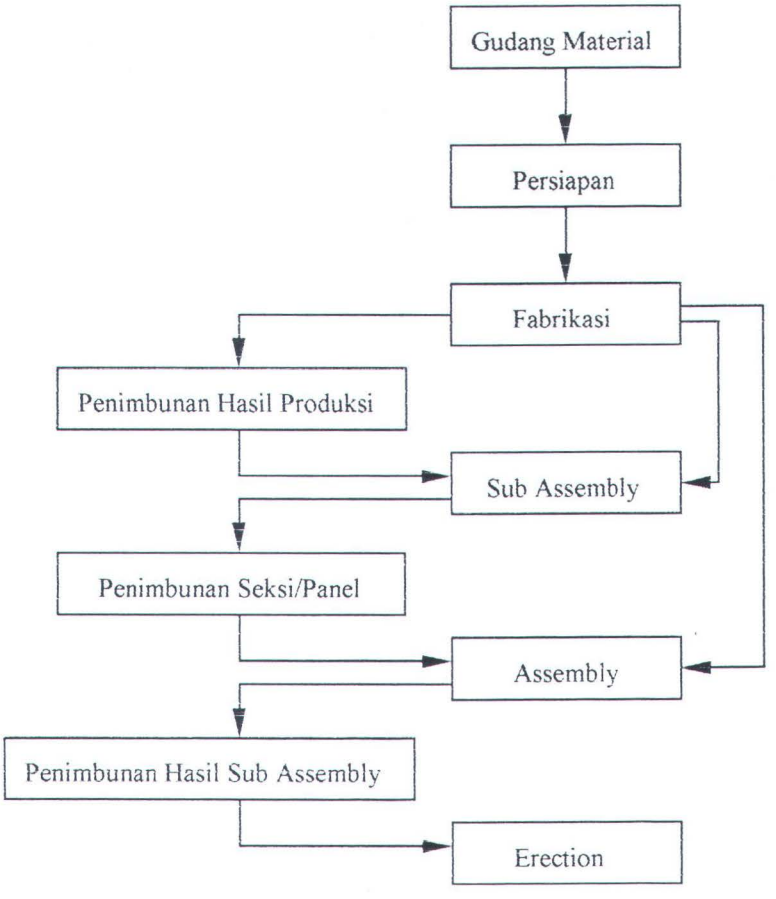
Dalam penulisan sub bab ini akan diberikan contoh aplikasi dua aspek yang paling dominan dalam penyusunan *layout* galangan kapal. Kedua aspek tersebut yaitu bentuk/tipe *layout* galangan kapal dan urutan proses produksi. Berdasarkan tipe galangan seperti yang telah dijelaskan pada sub bab terdahulu dapat diimplementasikan pada gambar 2.5 dengan mengambil contoh *layout* tipe L.



Gambar 2.5. Representasi *layout* galangan kapal tipe dasar L

Urutan proses produksi merupakan aspek paling penting dalam perencanaan *layout* galangan, karena menyangkut aspek-aspek yang lain yaitu hubungan antar fasilitas/kompartemen, material handling, derajat kedekatan dan lain-lain. Urutan proses produksi yang baik disertai dengan peminimuman aliran material akan menghasilkan efektifitas kerja pada galangan kapal. Di bawah ini diberikan

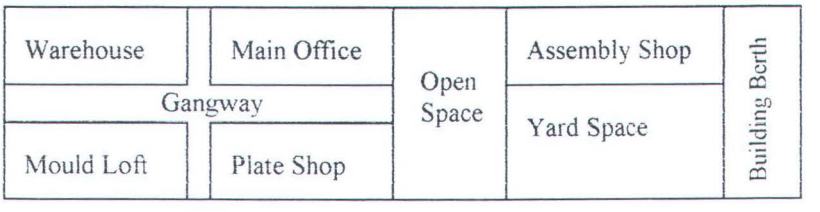
diagram tahapan proses produksi (Andjar Soeharto dan Soejitno, 1996) pada gambar 2.6 disertai implementasinya pada *layout* galangan kapal (gambar 2.7)



Gambar 2.6. Diagram tahapan proses produksi

Dalam gambar di atas, gudang material mempunyai kedekatan dengan persiapan, persiapan di sini berarti pekerjaan awal untuk proses selanjutnya yaitu proses fabrikasi. Pada representasi *layout* dibawah ini gudang material direpresentasilan oleh *warehouse*. Persiapan dalam layout direpresentasikan oleh *mould loft*, yaitu fasilitas untuk pekerjaan pembuatan rambu dan penandaan (*marking*). Fabrikasi direpresentasikan oleh bengkel pelat (*plate shop*),

penimbunan hasil produksi direpresentasikan oleh penimbunan sementara (*open space*), *sub assembly* direpresentasikan oleh *assembly shop*, penimbunan hasil *sub assembly* dan *assembly* direpresentasikan oleh tempat perakitan (*yard space*) dan *erection* direpresentasikan oleh *Building Berth*.



Gambar 2.7. Representasi *layout* galangan berdasarkan urutan proses produksi

2.3 KOMPUTER SEBAGAI SARANA PENDUKUNG

Semua kegiatan yang menyangkut dalam pemanfaatan desain lama memerlukan komputer untuk mendukung seorang desainer dalam menghasilkan desai-desain baru. Perkembangan dari sistem CAD dewasa ini tentunya memiliki kapasitas secara langsung dalam pemanfaatan desain-desain tersebut. Demikian pula pada sistem-sistem lain yang telah dihasilkan/diimplementasikan oleh program (*software*) tertentu telah banyak menunjang dalam pemanfaatan desain-dasain lama untuk proses desain baru (*design reuse*). Program komputer (model matematis, simulasi dan sebagainya) bagaimanapun merupakan alat yang sangat berguna dan mempunyai kekuatan besar dam membandingkan pilihan susunan wilayah kegiatan, dalam batasan kriteria yang terpilih dan data yang tersedia (Apple, 1977).

Dukungan komputer pada pemanfaatan desain-desain lama tersebut dapat diuraikan dibawah ini :

- Mendukung penyimpanan dan representasi desain-desain lama.
- Mendukung pengindeksan pada desain.
- Mendukung pencarian (*searching*) dari desain lama.
- Mendukung kesesuaian pola antara masalah desain yang akan dipecahkan dengan desain lama.
- Mendukung kesamaan pengukuran antara *input* dengan desain lama.
- Mendukung penyeleksian/pemilihan kesamaan terbanyak kepada *input*.
- Mendukung pengelompokan desain-desain.

Dengan demikian seorang desainer akan memiliki keuntungan lebih banyak jika menggunakan komputer sebagai alat (*tools*) untuk penyelesaian desainnya dibandingkan dengan cara manual yang dijelaskan pada sub bab sebelumnya. Dengan penggunaan komputer, desainer tidak perlu bingung dengan arsip desain yang dimilikinya karena telah disimpan dalam komputer dan jika sewaktu-waktu dibutuhkan dapat ditampilkan secara capat, rapi dan akurat. Hal ini tentu berbeda dibandingkan dengan cara penyimpanan manual. Lebih dari itu, dengan komputer proses pemilihan kesesuaian *input* desain baru dengan desain lama akan lebih banyak memberikan bantuan kepada desainer dan tentunya akan mempersingkat waktu penyelesaian. Semuanya itu akan berdampak positif bagi desainer karena dari keuntungan-keuntungan di atas telah didapatkan manfaat utama yaitu mengurangi beban kerja desainer.

ITS

2.4 *CASE BASED REASONING (CBR)* BIDANG DESAIN

Case based reasoning (CBR) adalah bagian dari *Artificial Intelligence* yang diperkenalkan oleh John S. Gero dan Fay Sudweeks. Beberapa contoh pengertian dari *CBR* antara lain disebutkan *CBR* merupakan pendefinisian sebuah model dalam *Artificial Intelligence* (Schank, 1982 ; Kolodner, 1985 ; Hammond, 1989 ; Slade, 1991). Kemudian menurut pengertian yang lain, *CBR* adalah model desain yang memfokuskan penggunaan kembali pengalaman desain tertentu peristiwa peristiwa terdahulu (Maher and Zhang, 1991). Peristiwa-peristiwa tersebut berhubungan dengan kasus desain terdahulu. Dalam model ini, pertimbangan analogi *(analogical reasoning)* digunakan untuk menyeleksi dan mentransformasikan hasil-hasil desain lama tertentu menjadi penyelesaian masalah untuk desain baru. *Analogical reasoning* tersebut mengacu pada dasar pemikiran dari sebuah ide bahwa kasusu-kasus atau pengalaman-pengalaman terdahulu memberikan pengetahuan dan bantuan dalam memecahkan masalah baru (Maher dkk., 1995). Sebagai contoh, ketika seorang desainer mendesain layout galangan kapal maka akan mengingatkan dia pada desain layout sebuah pabrik. Sedangkan menurut Maher and Gomez (1996), *CBR* merupakan model umum untuk dasar pemecahan masalah dari pemanggilan dan penarikan atas pengalaman tertentu. Model tersebut telah diaplikasikan dan diperluas dalam berbagai tingkatan pada masalah desain. Batasan dan tipe dari penerapan desain ditunjukkan pada penggunaan model *CBR* dan memperlihatkan bagaimana penerapan tersebut dapat memecahkan persoalan desain.

Teori ini adalah suatu pendekatan untuk menyelesaikan permasalahan dengan memperhatikan kasus yang ada dimana keduanya memiliki kesamaan situasi. Dari kasus-kasus terdahulu akan diambil metode pemecahannya dan diimplementasikan ke dalam permasalahan yang baru, sehingga bisa diperoleh pemecahannya. Dari hasil yang diperoleh dapat digunakan sebagai pendekatan untuk menyelesaikan permasalahan baru atau dapat dimasukkan ke dalam *case base* sebagai referensi untuk problem selanjutnya.

Maher dkk. (1995) memperkenalkan sebuah model sederhana yang ditunjukkan pada gambar 2.8. Pada model ini, desain-desain lama direpresentasikan oleh sebuah tempat desain (*case base*). Jika diberikan masalah baru sebagai input, maka desain lama akan ditarik dan selanjtnya diadaptasi untuk menghasilkan desain baru. Untuk lebih jelasnya langkah-langkah dalam proses pemodelan pada *CBR* dapat diterangkan sebagai berikut:

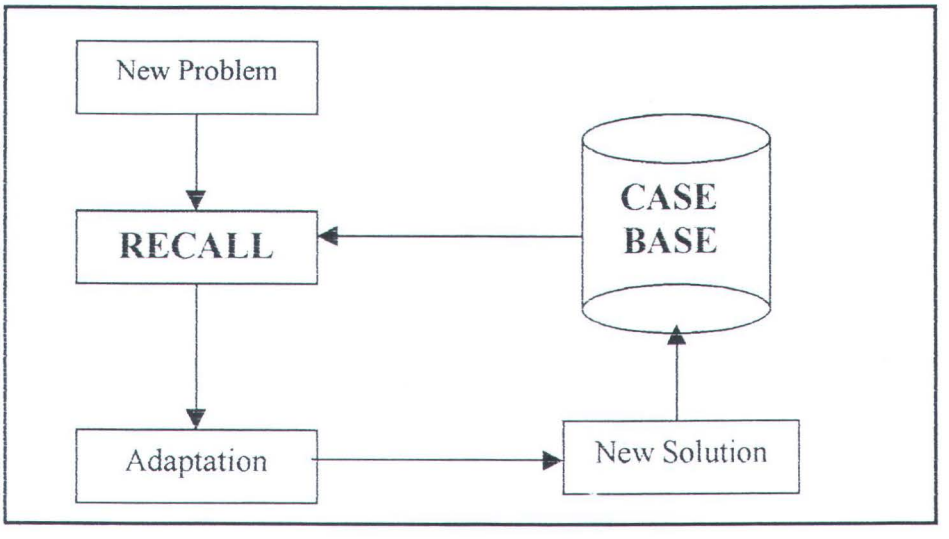
1. Adanya permasalahan baru *(new problem)*.
2. Dengan adanya masalah baru, desainer mengumpulkan desain yang sudah ada kedalam *case base*. Desain yang dimasukkan memiliki kesamaan situasi, fungsi dan karakteristik dengan masalah yang akan dipecahkan.
3. *Recall*, tahap penarikan salah satu desain dari *case base*.

   Tahapan dalam *recall* adalah :

   - Pencarian *(searching)*
   - Kesesuaian *(matching)*
   - Pengukuran kesamaan *(measuring similarities)*
   - Penarikan *(retrieving)*

- Penyeleksian *(selection)*

4. *Adaptation*, tahap modifikasi desain yang diperoleh dari tahap *recall*.
5. Diperoleh desain baru *(new solution)* yang sesuai keinginan *owner*.
6. *New solution* dapat digunakan atau disimpan kembali dalam *case base*.



Gambar 2.8. Model *case based reasoning* sederhana dari Maher dkk (1995).

2.4.1 REPRESENTASI KASUS DESAIN

Kolodner (1994) menyatakan bahwa kasus adalah pengalaman yang harus dipelajari. Manfaat dari sebuah kasus yang dipelajari dari sebuah kasus adalah untuk membantu menyelesaikan problem yang relevan di waktu yang akan datang.

Representasi kasus desain dalam *case base* memberikan dasar untuk pemanfaatan kembali kasus desain tersebut oleh sistem *CBR* dan mendefinisikan aplikasinya dalam *case base* (Maher dkk., 1995). Dengan kata lain efisiensi dari proses penarikan kasus desain tergantung dari representasi kasus dalam *case base*.

Pada proses penarikan kasus (*recall*), akan dilakukan pencarian kasus-kasus desain dengan situasi relevan dengan kasus desain yang akan dipecahkan, hingga didapatkan desain baru yang diinginkan (*new solution*). Pada proses adaptasi, dibuat keputusan pada bagian desain mana yang perlu dirubah dan bagian mana yang dibiarkan tetap sama dengan desain lama (Maher dkk., 1995).

Menurut Maher dkk. (1995), pengorganisasian *case base* meliputi

1. Isi kasus desain.
2. Representasi isi kasus desain.
3. Pengorganisasian kasus desain dalam *case base*.

Isi Kasus Desain

Isi dari kasus desain mempunyai beberapa bentuk yang berbeda, termasuk situasi kasus terdahulu, persyaratan dan penyelesaian kasus terdahulu, perbedaan sudut pandang terhadap kasus dasain dan juga proses penyelesaiannya (Maher and Garza, 1997).

Situasi kasus dasain terdahulu dapat digunakan oleh desainer untuk mempelajari keadaan baik atau buruk dari kasus dasain terdahulu serta penyelesaiannya, sehingga situasi kasus tersebut dapat membantu desainer dalam mengambil keputusan untuk menyelesaikan kasus yang baru. Sistem ARCHIE, ARCHIE-II dan Gross dkk. adalah contoh dari sistem yang menggunakan situasi kasus desain terdahulu. Sistem ARCHIE-II dikembangkan sebagai lanjutan dari sistem ARCHIE. Kedua sistem tersebut memfokuskan pada keinginan desainer untuk mencari representasi grafik dari desain arsitektur dan meggunakan situasi

kasus desain terdahulu untuk melakukan hal tersebut. Sistem Gross dkk. bergabung dengan ARCHIE-II sebagai bagiannya. Sistem ini digunakan untuk mengenali penggambaran sketsa tangan dan diagram yang ditetapkan desainer dimana hasil hasil dari proses pengenalan tersebut digunakan untuk mendapatkan bentuk penyelesaian dari sistem ARCHIE-II.

Persyaratan-persyaratan pada kasus desain dapat digunakan sebagai parameter dalam penarikan kasus. Syarat-syarat tersebut dapat langsung dibandingkan dengan syarat yang berlaku pada masalah desain yang akan dipecahkan. Penyelesaian dari sebuah kasus terdahulu dapat digunakan sebagai acuan awal dalam menyelesaikan masalah baru, dimana penyelesaian tersebut dapat menjelaskan lebih detail tentang masalah tersebut. Untuk penyelesaian kembali kasus desain, penyelesaian kasus desain terdahulu disusun secara khusus dalam sebuah nama file yang disimpan dalam *case base* (Maher dkk., 1995).

Dalam beberapa sistem digunakan kombinasi antara persyaratan kasus desain terdahulu dan penyelesaiannya sebagai isi dari *case base*. Sistem ARCHIE, ARCHIE-II dan Gross dkk. menggunakan penyelesaian kasus desain terdahulu untuk menyediakan informasi bagi desainer mengenai bagian dari penyelesaian masalah, termasuk pemikiran tentang hal-hal yang berhubungan kasus desain terdahulu. CADRE dan FAMING menggunakan kombinasi dari penyelesaian dan persyaratan dari desain-dasain lama yang telah ada. Dalam kedua sistem tersebut penyelesaian kasus desain direpresentasikan sebagai rencana dasar dan persyaratan desain mengambil bentuk batasan dalam dimensi rencana dasar. Perbedaan pendapat dari masalah desain mengacu pada perbedaan representasi

dari kasus desai terdahulu. Sistem ASA, FABEL dan Dave dkk. menggunakan bentuk ini sebagai isi dari permasalahan desain. Dalam sistem ASA (*Architectural Symbol Assistant*), untuk merepresentasikan kasus desain digunakan penyelesaian desain dengan berdasarkan pada teori yang menjelaskan karakteristik dari kasus tersebut. Ada dua jenis teori untuk menjelaskan hal diatas yaitu: *accessibility theory*, yang merepresentasikan *topological relation* antara ruangan dalam sebuah layout berdasarkan kedekatannya dan *night-zone-distribution theory*, yang merepresentasikan hubungan pertukaran satu ruangan dengan ruangan lainnya dalam sebuah layout. Dalam FABEL, perbedaan pendapat dari permasalahan desain termasuk vektor ciri-ciri (dartar nilai dari ciri-ciri permasalahan), gambaran penyelesaian permasalahan desain (digambarkan dengan rangkaian bitmaps), gestals (pendapat umum tentang hubungan antar ruang) dan hubungan topologi (*topological relation*) antar ruang. Dalam sistem lainnya yaitu Dave dkk., kasus desain direpresentasikan dalam gambar geometri (*geometrical drawing*) yang dilengkapi dengan informasi yang berupa simbol dan *topological relation* antar ruangan dalam layout berdasarkan kedekatannya.

Representasi Isi Kasus Desain

Representasi isi dari kasus desain tergantung pada bentuk dari isi kasus desain dalam *case base*. Bentuk dari isi yang umum digunakan adalah *Attribute-value pairs* (Kolodner, 1993; Giretti dkk., 1994; Voβ dkk., 1994; Maher dkk., 1995; Flemming dkk., 1997) sebagai attribut masing-masing mempunyai nilai.

Bentuk lainnya adalah *Text* (Domeshek and Kolodner, 1992; Kolodner, 1993; Gross dkk., 1994), *Mathematical-equalities and inequalities* (Hua dkk., 1996; Faltings, 1997), *Graph-based representation* (Giretti dkk., 1994; Voβ dkk., 1994; Dave dkk., 1994), Symbol/icon (Giretti dkk., 1994; Voβ dkk., 1994) dan *Multimedia representations* (Domeshek and Kolodner, 1992; Gross dkk., 1994).

Kasus desain yang direpresentasikan dengan *Attribute-value pairs* digambarkan dengan attribut, yang masing-masing mempunyai nilai. Contohnya adalah sebuah desain ruang akomodasi dapat dijelaskan dengan informasi geometris (panjang dan lebar). ASA memakai bentuk di atas untuk merepresentasikan layout, FABEL menggunakannya untuk merepresentasikan persyaratan desain, sedangkan SEED menggunakannya untuk menjelaskan karakteristik obyek dalam kasus desain.

Text digunakan untuk merepresentasikan situasi kasus (Kolodner, 1993) yang menggambarkan struktur kasus desain. ARCHIE dan Gross dkk. menggunakan bentuk ini.

Mathematic equalities and inequalities digunakan untuk merepresentasikan design constrain dengan kasus desain.

Graph digunakan untuk merepresentasikan hubungan antara attribut-attribut kasus desain, dimana atribut tersebut ditampilkan dalam bentuk *node* dan garis penghubung antara dua *nodes*. ASA, FABEL dan Dave dkk. menggunakan bentuk ini untuk menggambarkan kedekatan antara ruangan dalam sebuah layout.

Symbol/icon digunakan dalam merepresentasikan analisa tertentu pada layout. Dalam ASA, symbol digunakan untuk merepresentasikan *night zone theory*.

Multimedia representation merepresentasikan kombinasi dari sejumlah kasus desain. Gross dkk. menggunakan bentuk ini untuk mempermudah pemahaman terhadap permasalahan dengan memberikan beberapa model pemahaman.

Adanya kompleksitas dalam kasus desain, kasus dapat dibagi menjadi beberapa bagian yang lebih spesifik, misalnya untuk problem *layout* ruang akomodasi akan dipisahkan menjadi beberapa problem yang lebih khusus yaitu interior, persyaratan ruangan dan kedekatan antar ruangan.

Pengorganisasian Kasus Desain dalam *case base*

Pengorganisasian kasus desain dalam *case base* dilakukan dalam bentuk *Indexing scheme* (Maher dkk., 1997). Dalam bentuk ini setiap kasus desain akan diindeks secara individual dan kemudian disimpan dalam *case base* dalam daftar (solusi, proses penyelesaian, permasalahan). Bentuk lain dari *indexing scheme* adalah mengindeks klas dari kasus desain berdasarkan kesamaan tertentu. Tujuan dari hal ini adalah untuk mengurangi jumlah kasus desain yang dicari untuk memperoleh relevansi dengan kasus yang akan dipecahkan.

2.4.2 PENARIKAN KASUS DESAIN

Penarikan kasus desain berdasarkan kesamaan situasi permasalahan antara kasus desain baru dengan kasus desain dalam *case base*. Penarikan kasus desain ini dapat dilakukan dengan menggunakan metode penarikan kasus desain. Metode ini dibagi atas dua kategori yaitu :

1. *User directed layout case selection*

   Desainer (*user*) menarik kasus secara langsung dengan mengabaikan proses *pattern matching*. Relevansi antara desain problem dengan kasus desain dalam *case base* adalah berdasarkan dari desainer.

2. *Pattern Matching based layout case retrieval*

   Penarikan melalui *pattern matching* akan melewati proses pencarian kasus, penentuan kesamaan kasus dengan desain problem dan kemudian memilih kasus yang relevan.

Pattern Matching merupakan inti dari proses dalam kategori ini (Maher dkk., 1995). Dengan adanya *pattern matching*, pola permasalahan dari desain problem akan sesuai dengan pola pada kasus dalam *case base*. *Symbolic pattern matching* digunakan untuk proses *matching* atribut desain *layout* dalam bentuk simbol, sedangkan *topological pattern matching* digunakan untuk *matching* pada struktur dalam layout. Dalam *topological relations*, digunakan *graph matching technique* yang memungkinkan untuk menarik hubungan kedekatan antar ruangan pada kasus desain terdahulu dengan *input* berasal dari desainer.

2.4.3. ADAPTASI KASUS DESAIN

Adaptasi kasus desain dititikberatkan dengan memodifikasi kasus terpilih dari *case base* untuk mendapatkan penyelesaian desain baru yang memenuhi persyaratan desain baru. Ada beberapa metode adapatasi kasus yang dapat dibedakan ke dalam dua kategori yaitu: informal dan formal.

Metode Informal mengacu pada metode yang dalam permasalahan proses adaptasi ini dibawa langsung oleh desainer. Gross system menggunakan metode ini untuk menyajikan sebuah *browser case*. Yang menekankan penyampaian informasi tentang kasus desain kepada desainer yang lebih baik daripada adaptasi kasus desain yang diberikan kepada desainer.

Metode Formal mengacu pada metode yang menggunakan alogoritma untuk menunjang adaptasi otomatis kasus desain. Metode ini terdiri dari batasan pemenuhan persyaratan, teknik perubahan topologi, penggerakan dan pengujian, perencanaan ulang desain. Metode batasan pemenuhan persyaratan diterapkan untuk adaptasi kasus yang digambarkan sebagai permasalahan batasan pemenuhan persyaratan (Maher dkk., 1995; Hua dkk., 1996). Oleh karena itulah, adaptasi sebuah kasus desain bertujuan untuk memenuhi seperangkat batasan yang diinginkan dari permasalahan desain yang baru.

Jenis-jenis adaptasi kasus di mana diterapkan metode batasan pemenuhan persyaratan meliputi adaptasi dimensional dan adaptasi desain obyek dari permasalahan desain *layout*. Adaptasi dimensional mengacu pada perubahan dimensi kasus desain *layout*. Sistem FABEL mengggunakan Metode batasan

pemenuhan persyaratan untuk menyelesaikan permasalahan penempatan desain obyek dalam kasus desain *layout* yang diadaptasi.

Teknik-teknik perubahan topologi digunakan untuk adaptasi topologi yaitu: modifikasi dari hubungan topologi antar elemen kasus desain *layout*, seperti hubungan kedekatan antar ruangan. Dave system menggunakan peraturan perubahan topologis yang mengungkapkan bentuk pengetahuan tentang manipulasi elemen dan hubungan dari kasus desain *layout* sehingga penyelesaian *layout* yang diadaptasi dengan tepat dapat diperoleh. Dalam FABEL system digunakan metode mencoba dan menghasilkan, dimana akan diberikan susunan layout dan penarikan kembali kasus desain *layout* dievaluasi untuk mengetahui ketepatan penempatan obyek tidak bertentangan dengan elemen dari kasus desain dalam *layout*. Bila terjadi perbedaan, penempatan obyek desain akan dimodifikasi berdasarkan kasus desain. Perubahan ini akan digunakan untuk menyelesaikan kasus desain yang lain. Proses ini dilakukan sampai penyelesaian *layout* yang sesuai persyaratan didapatkan.

2.5 PENDEKATAN *PATTERN MACHING*

Aspek utama dalam CBR adalah *pattern matching* (bentuk kesamaan), *pattern matching* dipakai untuk menentukan tingkat relevansi antara input dengan referensi yang ada di dalam *case base*. Aspek ini menunjukkan tingkat kesamaan antara permintaan dan lingkup matching serta membantu dalam proses pattern matching hingga selesai. Dengan menganalisa *pattern matching* akan diperoleh desain yang memiliki kemiripan.

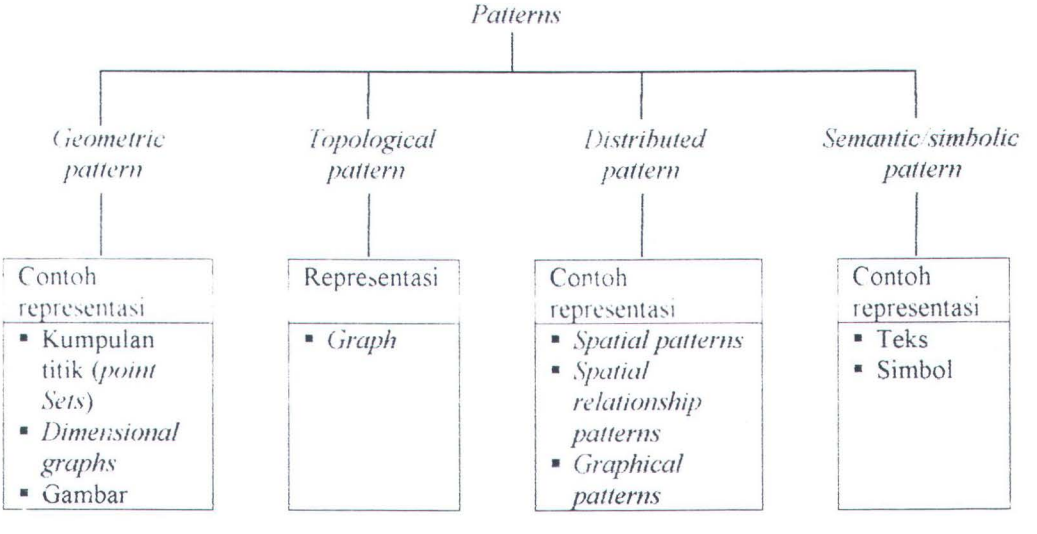
Tiga macam karakteristik dari *pattern matching* ditunjukkan sebagai berikut:

1. Klasifikasi *Pattern*
2. Metode *Matching*
3. Jenis Kesamaan (*similarity*)

2.5.1 KLASIFIKASI *PATTERN*

Pattern matching dapat diklasifikasikan menjadi empat bagian utama (Manfaat dkk., 1996):

1. *Geometric Pattern*, mempresentasikan bentuk dalam gambar, *point* dan dimensi *graph.*
2. *Topological Relations*, mempresentasikan bentuk dalam jaringan atau *graph* yang tersusun dan saling berhubungan antara elemen-elemen desain.



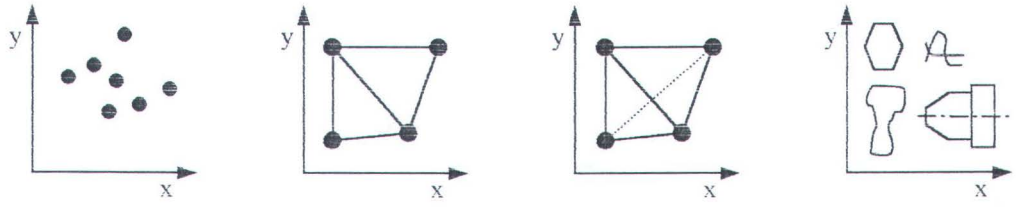
Gambar 2.9. Empat bagian dasar dari *patterns* (Manfaat dkk., 1996).

3. *Distributed Pattern*, mempresentasikan bentuk dalam *spatial pattern, spatial relation* dan *graph pattern.*
4. *Semantic Symbolic Pattern*, mempresentasikan dalam bentuk teks dan simbol

Pembagian ini menunjukkan adanya perbedaan dasar pada representasi *pattern* yang digunakan dalam teknik *pattern matching*. Penjelasan dari bagian-bagian dari *pattern matching* dapat diterangkan secara singkat sebagai berikut :

Geometric Pattern

Pola-pola pada teknik ini didefinisikan dalam sistem koordinat. Kumpulan titik (*point sets*), sebagai salah satu contoh yang direpresentasikan dalam kelas *geometric pattern* yang mana dapat didefinisikan sebagai kumpulan dari koordinat. Pada gambar 2.10a diberikan contoh dari kumpulan point dalam sistem koordinat dua dimensi (2D). *Dimensional Graphs* menunjukkan *graph* yang memiliki elemen-elemen *vertices, edges dan faces* yang didefinisikan dalam sistem koordinat. *Graph-graph* tersebut dapat berbentuk dua dimensi *(planar graphs)* atau tiga dimensi *(polyhendra)*. Gambar 2.10b merupakan contoh dari *planar graph* dimana *vertices* adalah point dalam koordinat sistem 2D. Sedangkan gambar 2.10c adalah contoh dari *polyhedron* sebagai representasi *dari three dimensional (3D) graph*. Contoh dari gambar-gambar dalam *kelas geometric pattern* terdapat bermacam-macam tipe dalam sistem koordinat. Tipe dari gambar-gambar tersebut dapat berupa *lines, arcs, circles, polygons, curves dan characters*. Gambar 2.10d merupakan variasi tipe gambar dalam sistem koordinat 2D.



(a) Kumpulan point dalam sistem koordinat 2D

(b) *Planar graph* dasistem koordinat 2D

(c) *Polyhedron* direpresentasikan dalam 3D graph

(d) Variasi tipe gambar dalam sistem koordinat 2D

Gambar 2.10. Ilustrasi dari contoh gambar pada kelas *geometric pattern* (Manfaat dkk., 1996).

Topological Relations

Pada kelas *topological relations* merepresentasikan pola-pola *(patterns)* dalam hubungan kedekatan antara elemen-elemen atau bagian dari pola-pola. Kelas ini direpresentasikan sebagai *graph* yang dibentuk oleh *vertices, edges* dan *faces. Vertices* merepresentasikan elemen-elemen atau bagian dari pola, *edges* merepresentasikan hubungan antara elemen-elemen tersebut. Sedangkan *face* dibentuk berdasarkan kumpulan kedekatan dari *edges* dan dalam banyak kasus hal tersebut dipergunakan untuk menunjukkan *graph matching*. Teknik ini dikenal dengan nama *topological pattern (graph) matching*. Penjelasan secara rinci mengenai teknik *topological pattern (graph) matching* dapat dilihat pada bab selanjutnya disertai dengan contoh penggunaannya.

Distributed Pattern

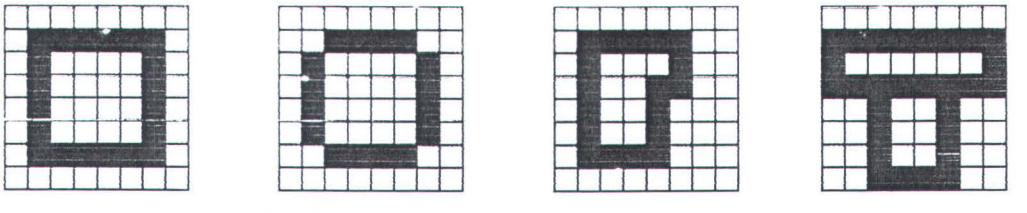
Istilah "*distributed pattern*" dikemukakan untuk menunjukkan pendistribusian pola-pola dalam bentuk matriks *(grid units* atau *pixels)* dalam

layar monitor. Contoh dari representasi dalam kelas ini adalah *spatial patterns, spatial relationship patterns dan graphical patterns.*

Spatial patterns menghubungkan pola-pola yang memiliki bentuk *shapes* dari *spaces*. Masing-masing *grid unit* atau *pixel* dari *pattern* merepresentasikan bagian *shape* dari *space*. Gambar 2.11 memberikan macam-macam contoh dari *spatial pattern* yang mempunyai bentuk *pixel* 8 x 8.

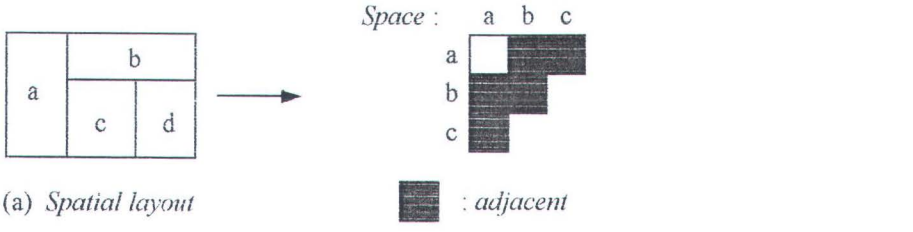
Spatial relationship pattern menghubungkan pola-pola yang merepresentasikan hubungan antara *spaces* pada *spatial layout* seperti yang diilustrasikan pada gambar 2.12. Pada pola ini beberapa *grid unit* atau *pixel* merepresentasikan hubungan kedekatan antara dua ruang dan juga dapat merepresentasikan posisi dari ruang tertentu yang memiliki kedekatan.

Graphical patterns menghubungkan beberapa pola lainnya *dari spatial patterns* atau *spatial relationship patterns*. Contoh dari *graphical pattern* tidak seperti yang diberikan pada gambar 2.11, yaitu pada *spatial pattern*. Pada dasarnya kelas ini merepresentasikan beberapa *grid unit* atau *pixel* yang merepresentasikan bagian dari pola, tetapi tidak dalam bentuk *spatial pattern*.



(a) *square pattern* (b) *O-shaped pattern* (c) *L-shaped pattern* (d) *T-shaped pattern*

Gambar 2.11. Macam-macam contoh dari *spatial pattern* yang mempunyai bentuk *pixel* 8 x 8 (Manfaat dkk., 1996).



(a) *Spatial layout*

(b) Representasi *spatial relationship pattern* dalam bentuk *pixel* dari *spatial layout* pada gambar (a)

Gambar 2.12. Contoh representasi *pixel* pada *spatial relationship pattern* (Manfaat dkk., 1996).

Semantic/Symbolic Pattern

Kelas *semantic/symbolic* pattern merepresentasikan atribut-atribut *semantic/symbolic* dari objek meliputi nama, fungsi, ukuran, warna dan atribut lainnya. Atribut-atribut dapat direpresentasikan dalam bentuk teks atau simbol (seperti dalam diagram, *icon* dan lainnya). Representasi dalam bentuk teks dapat digabungkan dengan nilai-nilai dimana bentuk pasangan nilai-nilai atribut merupakan representasi dari objek. Pasangan nilai-nilai atribut dalam banyak kasus direpresentasikan dalam bentuk daftar (*list*). Dalam bentuk simbol sebaiknya direpresentasikan dalam bentuk *list*.

Sistem dari Dave, dkk. (1994), Giretti, dkk. (1994) dan Gross, dkk. (1994) yang mana semuanya berlandaskan pada *CBR* dan Voβ, dkk. (1994) yang memfokuskan pada *access* dan penarikan desain menggunakan kelas ini untuk melakukan *pattern matching* sebagai bagian dari mekanisme penarikan desain. Pada Giretti, dkk., klas ini digunakan dalam penambahan untuk representasi

planar graph yang mana merepresentasikan *spatial relationship dari spatial layout*. Pada Voβ, dkk., digunakan dalam penambahan *planar graph* yang merepresentasikan *spatial relationship* dan *spatial pattern* dari *spatial layout*. Sedangkan pada sistem Gross, dkk., *symbolic pattern* digunakan dalam bentuk diagram dan dihubungkan dengan program komputer yang menginterpretasikan sketsa gambar tangan dan diagram-diagram. Sketsa ini dibentuk dari *kumpulan primitive elemen* yang disebut *glyphs*. Contoh dari *glyhps* adalah *lines, circles, boxes, triangels, arrows, numbers, letters* dan lain-lain.

Dalam penulisan ini penulis menggunakan teknik *Topological Relation yaitu topological pattern (graph) matching* untuk menganalisa pelaksanaan proses penarikan desain. Alasan penulis memilih teknik ini dalam menjalankan proses *pattern matching* dikarenakan teknik ini mudah dimengerti oleh pengguna dan cocok diterapkan pada desain *layout* galangan kapal karena berdasarkan kedekatan antar ruang yang diperlukan pada perencanaan *layout* galangan kapal yang dapat diartikan sebagai kedekatan ruang berdasarkan proses produksi sehingga dapat meminimumkan pergerakan material.

2.5.2 METODE *MATCHING*

Metode *matching* berdasarkan pada metode yang digunakan dalam proses *pattern matching* dengan tujuan untuk menentukan tingkat kesamaan antara pola-pola yang ada. Dalam proses *matching*, hal itu menunjukkan hubungan antara input dari luar dengan data yang ada di dalam *case base*. Hasil dari proses ini adalah tingkat kesamaan antara pola-pola yang ada.

Metode digunakan untuk menganalisa pendekatan *pattern matching* dan itu akan menunjukkan adanya metode yang berbeda dalam penggunaan tipe *patterns* dan jenis kesamaan yang berbeda-beda. Ada tiga metode yang biasa dipakai yaitu: *neural networks, mathematical technique* dan *symbolic pattern matching.*

Neural networks

Metode ini merupakan model matematis dari struktur otak manusia (Coyne dan Newton, 1989), disebut juga *parallel distributed processing system* (Rumelhart, 1989). Metode ini terdiri dari sejumlah jaringan unit proses (*nodes*) yang berisi data dan dihubungan dengan garis penghubung (*weighted arcs*). Dalam prosesnya input akan di*matching* dengan sejumlah data yang sudah ada dan dengan proses iterasi akan diperoleh kesamaan antara input dengan data.

Metode ini mampu mengenali pola-pola yang tidak lengkap atau tidak teratur dan menghasilkan pola baru dengan pola yang lengkap. Kemampuan ini sangat mendukung dalam pengenalan desain, kontrol proses dan lainnya.

Mathematical technique

Metode ini banyak digunakan dalam kasus yang mengandung *geometric* dan *graph matching.* Pada intinya metode ini menggunakan teori matematika untuk menganalisa kesamaan antar pola-pola yang berbeda. Beberapa metode dalam matematika teknik adalah: *coordinat distance measurement, hashing method* (Akutsu, 1995), *dynamic matching method* (Akutsu, 1995), *cross*

correlation (Cooper,1989) dan *graph theoretic algorithms* (Hopcroft dan Tarjan, 1973).

Coordinat distance measurement

Metode ini digunakan untuk mengukur jarak antara titik, garis, kurva dan lainnya dalam sistem koordinat. Schneider dkk (1989), menggunakan metode ini untuk mengukur kesamaan bentuk antara dua gambar.

Hashing method

Metode ini memberikan akses langsung dan cepat pada data tertentu dengan cara memberi nilai tertentu dalam data base.

Dynamic matching method

Digunakan dalam proses *matching* bentuk tiga dimensi yang merupakan kombinasi antara *dynamic programming technique* dengan Kabsch's metode (1976). Tujuannya adalah meningkatkan efisiensi proses *pattern matching*.

Cross correlation

Berdasarkan pada *method template matching* yang mengukur kesamaan antara template dengan data/gambar.

Graph theoretic algorithms

Metode ini berdasarkan pada algoritma untuk mencari kesamaan antara *graph*. Ada beberapa tipe kesamaan antar *graph* (Harary, 1969 dan Berger, 1976) :

1. *Graph isomorphism* : mencari kesamaan antara *graph* dengan *graph* lainnya.

2. *Subgraph isomorphism* : mencari kesamaan antara *graph* dengan *subgraph* dari *graph* lainnya.
3. *Double subgraph isomorphism* : mencari kesamaan antara *subgraph* sebuah *graph* dengan *subgraph* dari *graph* lainnya

Ballard dan Brown (1982), memberikan contoh dari ketiga tipe diatas yaitu: *matching metric*, *backtrack search* dan *associated graph matching* (pembentukan struktur data dari dua *graph* yang dilakukan *matching*).

Symbolic pattern matching

Dipakai untuk proses matching dimana obyek mempunyai pola berupa simbol-simbol. Dalam prosesnya, pola-pola itu akan direpresentasikan sebagai simbol.

Untuk metode matching dalam penulisan ini menggunakan *matematical technique* khususnya *associated graph matching* karena dapat menunjang pembentukan sebuah *graph* dari dua *graph* yang di*matching*.

II.5.3 JENIS KESAMAAN

Penentuan kesamaan dilihat dari tinjauan terhadap suatu obyek (Rips dkk., 1989). Tinjauan dapat berupa kategori obyek dan cara penyelesaiannya. Kategori dapat meliputi atribut-atribut tertentu yang dimiliki oleh obyek (*properties*), atribut tersebut dapat berupa fungsi, bentuk, ukuran atau hubungan lainnya.

Smith dkk. (1989) mendefinisikan kesamaan kedalam dua tipe yang berbeda yaitu: *concrete-relation* dan *abstract-relation*. *Concrete-relation* adalah hubungan

kesamaan antar atribut yang dimiliki oleh obyek. Sedangkan *abstract-relation* adalah hubungan yang mengatur kesamaan antara obyek. Pembagian klasifikasi *concrete relation* ke dalam lima bagian: *resemblances, overall similarity, identity, part-similarity,* dan *part-identity.*

Resemblances

Klas ini merupakan hubungan yang meliputi seluruh hubungan kesamaan (Smith, 1989). *Resemblances* bersifat umum, contohnya adalah *layout* ruang akomodasi kapal dengan *layout* galangan kapal

Overall similarity

Dalam kesamaan ini semua hubungan yang terjadi dalam obyek dinyatakan dalam satu karakteristik. Contohnya adalah *layout* ruang akomodasi kapal Cargo adalah *overall similarity* dengan *layout* ruang akomodasi pada kapal penumpang, karena keduanya merupakan layout ruang akomodasi di kapal.

Identity

Pada kesamaan ini semua hubungan yang terjadi dinyatakan sebagai satu karakteristik dan harus identik dengan *input (template).* Perbedaan dengan *overall similarity* adalah pada tingkat kesamaannya. Contohnya adalah kriteria dalam chemical *pattern matching,* dimana setiap *sub-graph* dari dua *graph* yang diuji harus identik.

Part-similarity

Pattern matching dalam kesamaan ini menggunakan sebagian dari atribut yang dimiliki oleh obyek. Contohnya adalah proses pencarian kesamaan dalam tugas akhir ini, dalam sub-graph setiap hubungan korespondensi *template* akan dipetakan ke dalam data *graph*. Jika salah satu hubungan memenuhi *part-similarity* maka akan diperoleh sebuah angka kesamaan.

Part-identity

Klas ini hampir sama dengan *part-similarity*, namun setiap atribut yang dilakukan matching harus identik antara *template* dan data. Contoh part-identity dengan tinjauan dari segi fungsi pada dua *layout* galangan kapal adalah identik.

BAB III

TEKNIK IMPLEMENTASI DARI *CBR*

BAB III

TEKNIK IMPLEMENTASI DARI *CBR*

Dalam bab ini akan dijelaskan representasi data dari desain *layout* lama dan *layout input* sebagai atribut yang dipakai pada proses *pattern matching* yang terdiri dari dua bagian, yaitu representasi ruang dan *layout*, kemudian teknik yang dipakai dalam proses pattern matching dan pendukungnya serta aplikasinya pada *layout* galangan kapal.

3.1 REPRESENTASI DATA

Untuk menjalankan proses penarikan kasus desain, semua hal yang berkaitan dengan *layout* harus didefinisikan dalam penyusunannya. Jika *layout* tersusun dari sejumlah ruangan, maka representasi tidak hanya untuk *layout* tetapi juga untuk ruangan yang membentuk *layout* tersebut. Berikut ini akan dijelaskan representasi dari *layout* dan ruang (*space*).

3.1.1 REPRESENTASI RUANG

Ruang direpresentasikan daiam sistem koordinat Cartesian dua dimensi yang terdiri dari dua bentuk utama yaitu: *Rectangle* dan *Polygon*. Ruangan didefinisikan sebagai ***Room***, yaitu ruangan yang merupakan bentuk asli dan menjadi penyusun *layout*. Pada *layout* galangan kapal, pengertian dari ruangan disamping berupa ruangan itu sendiri, dapat juga berupa tempat-tempat (*spaces)* atau fasilitas-fasilitas pada galangan kapal (facilities). Ruangan pada galangan

kapal mulai dari kantor *(office)*, bengkel-bengkel dan instalasi penunjang *(workshops and instalation)* sampai dengan fasilitas pokok galangan *(main facility)* seperti *floating dock, graving dock, building berth* dan lain sebagainya. Juga termasuk di dalamnya adalah *space* yang bukan merupakan ruangan seperti misalnya tempat peletakan material *(material stockyard)*, tempat perakitan *(yard space)* dan lain-lain. Sedangkan untuk pesawat angkat *(crane)* seperti *overhead crane, tower trav. Crane, floating crane* dan sebagainya dan alat transportasi yang dilengkapi dengan *crane (land transport equipment) seperti mobile crane, forklift* dan sebagainya tidak termasuk data yang dianalisa karena merupakan peralatan *(equipment)* yang bergerak *(movable)* sehingga tidak dapat dikategorikan sebagai ruangan dan hanya sebagai *access* saja.

Representasi dari *Room* tersusun dari beberapa attribut di bawah ini:

1. *Name*

   Setiap *Room* mempunyai pengenal yang berupa nama dari setiap ruang. Misalnya : Main Office, Electrical Shop, Building Berth, dan sebagainya.

2. *Text*

   Teks adalah label dari setiap *Room* dan ditampilkan dalam gambar *layout*.

   Misalnya :
   - Yard Space direpresentasikan dengan YSP
   - Power House direpresentasikan dengan POWER
   - Building Berth direpresentasikan dengan Berth

3. *Space-function*

   Setiap *Room* mempunyai fungsi.

Misalnya : fungsi dari gangway adalah tempat lalu lintas orang atau pergerakan material, maka bisa diberikan fungsi *access*.

4. *Space-identifier*

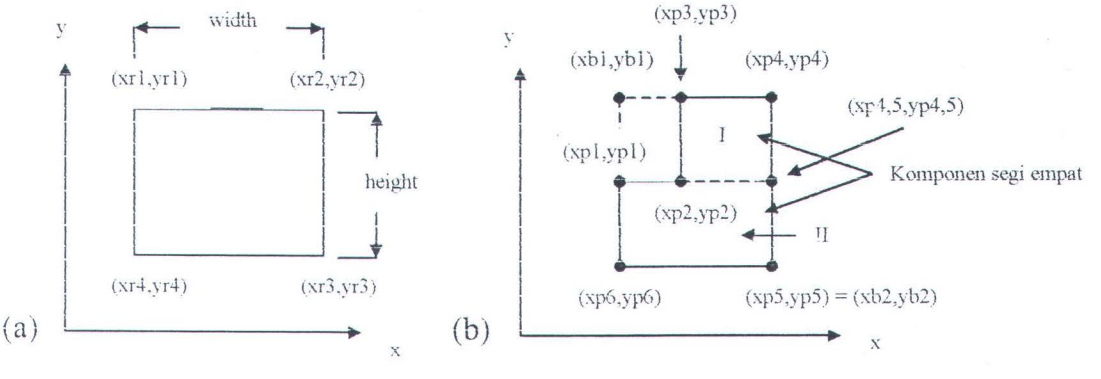
Identifikasi digunakan untuk membedakan *Room* akan masuk dalam kategori data atau *template*. Identifikasi ruang terdiri dari dua bagian yaitu: *stored-object* (untuk data) dan *template* (untuk *template*).

5. *Colour*

Digunakan untuk memberikan warna pada setiap ruang.

6. *Geometric shape*

Ada dua jenis *geometric shape* dari ruang yaitu: *Rectangle* dan *Polygon*. *rectangle* (segi empat) disusun oleh koordinat pojok kiri atas, lebar dan tinggi segi empat tersebut. Sedangkan penyusun *polygon* adalah titik-titik koordinat pembentuk dan segi empat penyusun poligon.



Gambar 3.1. Representasi ruang segi empat dan poligon (Manfaat, 1998).

Untuk bentuk segiempat dimulai dengan menentukan koordinat pojok kiri atas (xr1,yr1), lalu dilanjutkan dengan memasukkan panjang dan tingginya. Sedangkan untuk bentuk poligon terlebih dahulu ditentukan titik koordinatnya

(xp1,yp1), (xp2,yp2),...,(xp6,yp6). Setelah itu ditentukan segiempat penyusun poligon.

7. *Area*

Room memiliki luasan, dalam hal ini berfungsi hanya untuk memberi informasi.

8. *Bounding box*

Segi empat pembatas adalah batas maksimum dari sebuah segiempat ataupun poligon. Untuk segi empat pembatasnya adalah koordinat pojok kiri atas dan pojok kanan bawah, sedangkan untuk poligon adalah (xb1,yb1) dan (xp5,yp5) atau (xb2,yb2) pada gambar diatas.

9. *Vertices-coordinat*

Untuk menampilkan *Room*, diperlukan adanya titik-titik koordinat yang membentuk segi empat atau poligon. Segi empat mempunyai empat titik koordinat yaitu pada keempat sudutnya (xr1,yr1), (xr2,yr2), (xr3,yr3) dan (xr4,yr4). Sedangkan poligon titik koordinatnya adalah (xp1,yp1), (xp2,yp2),...,(xp6,yp6).

10. *Component rectangles*

Poligon dapat dibentuk dari gabungan beberapa buah segiempat. Segi empat itu disebut dengan *component rectangles*. Koordinatnya adalah (xp3,xp4,5), (yp3,yp4,5) untuk segiempat I dan (xp1,xp5), (yp1,yp5} untuk segiempat II.

3.1.2 REPRESENTASI *LAYOUT*

Representasi *layout* terdiri dari beberapa attribut sebagai berikut:

1. *Name*

   Setiap *layout* mempunyai nama untuk identifikasi. Misalnya, *Shipyard-1*, *Shipyard-2*, *Shipyard-3* dan seterusnya.

2. *Spaces*

   Merupakan semua ruangan yang ada di dalam *layout*.

3. *Assignment*

   Assignment bertujuan menempatkan *layout* sesuai dengan *layout* yang dimaksud. Misalnya *layout Shipyard-1....Shipyard-n*, maka *layout* ini akan dialokasikan ke dalam *case base* data untuk *Shipyard-Layouts*. Demikian pula untuk *Shipyard-Templete-1....Shipyard-Templete-n* akan dialokasikan data untuk *Layout-Templates*.

4. *Spaces-Closeness*

   Layout dapat dipandang sebagai kumpulan dari ruangan-ruangan yang berdekatan. Dua ruangan yang berdekatan adalah ruangan yang berbatasan langsung.

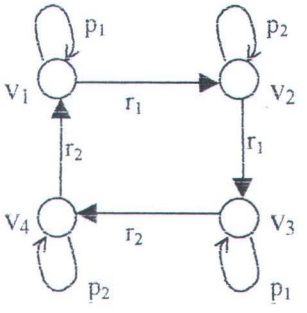
3.2 *TOPOLOGICAL PATTERN (GRAPH) MATCHING*

Topological pattern (graph) matching digunakan untuk mencari kesamaan hubungan antar ruang pada *layout* yang direpresentasikan dalam bentuk *graph*. Teknik ini memfokuskan kesesuaian hubungan objek pada *layout* yaitu pola-pola dalam ruang-ruang yang berdekatan secara keseluruhan. Sebagai catatan bahwa

dua ruang dikatakan berdekatan jika dua ruang tersebut diberikan batasan secara umum. Tujuan dari penggunaan teknik ini adalah untuk mencari jumlah *clique* terbanyak dari masing-masing *layout* yang disimpan dalam *case base* (Barrow and Burstall, 1976) serta digunakan juga sebagai teknik gabungan *graph* (Ballard and Brown, 1982).

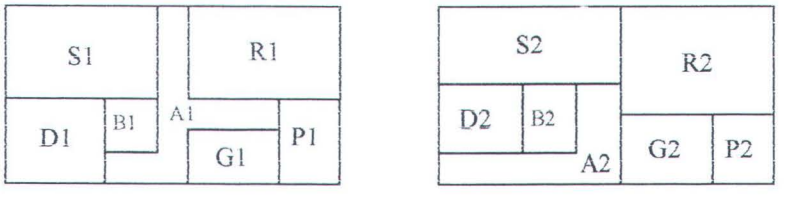
Teknik gabungan *graph* ini menggunakan data struktur yang disebut *"association graph"* dan *"clique finding algorithm"*. Proses seleksi dari teknik ini adalah kemampuan untuk mendapatkan kesamaan antara sub *graph* dari *graph* dan sub *graph* dari *graph* yang lain. Kemudian teknik ini akan memberikan kesesuaian terbaik antara dua *graph* yang didefinisikan sebagai perbandingan teknik untuk mendefinisikan kesamaan *graph* antara dua *graph* yang mana bertujuan untuk mendapatkan ketepatan kesamaan antara *graph* dimana *node* tidak terlepas dari satu atau *graph* yang lain.

Penjelasan teknik tersebut di atas adalah untuk menegaskan beberapa gabungan hubungan antar *graph*. Sebuah *graph* terdiri dari beberapa *node* v, beberapa properti p pada *node* dan beberapa hubungan r yang menghubungkan antara *node* yang satu dengan yang lain. Sebagai contoh representasi bentuk *graph* diberikan pada gambar 3.2. *Node* v_1 dan v_3 memiliki properti p_1, *node* v_2 dan v_4 memiliki properti p_2, antara node v_1 dan v_2 dan antara *node* v_2 dan v_3 masing-masing memiliki hubungan r1 dan antara *node* v_3 dan v_4 dan antara node v_4 dan v_1 masing-masing memiliki hubungan r2.

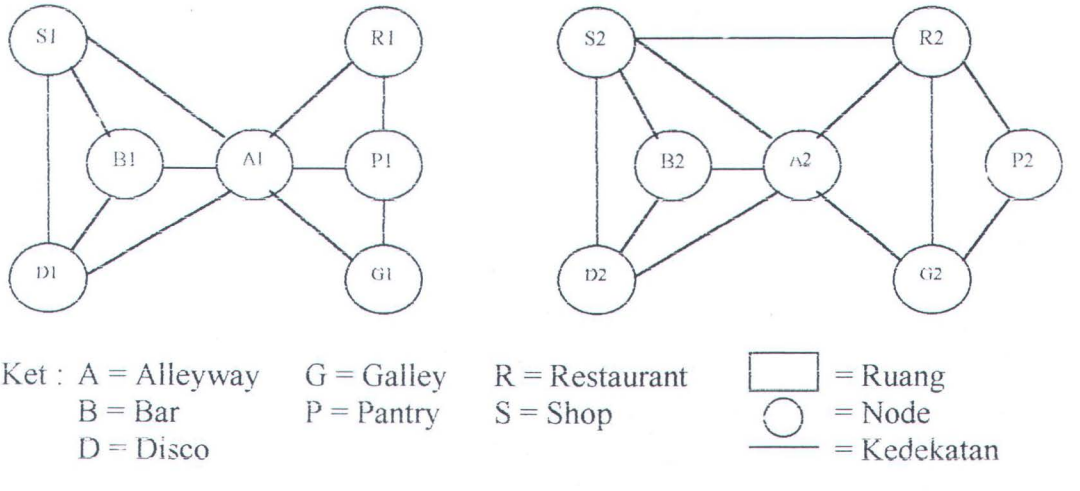


Gambar 3.2. Representasi sebuah *graph* dengan beberapa properti dan hubungan (Manfaat dkk., 1998).

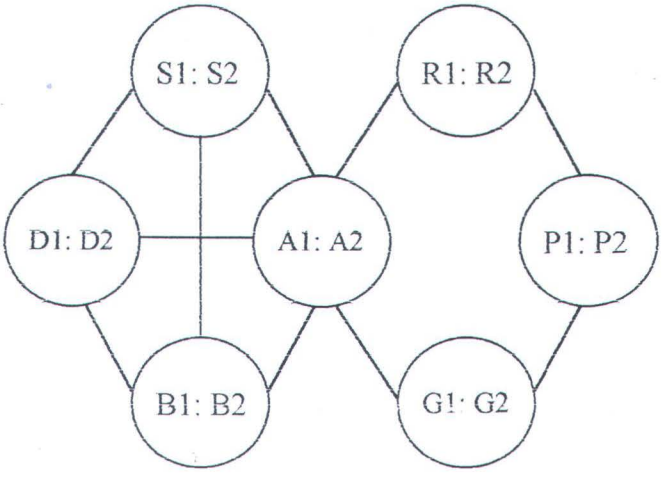
Sebagai contoh bentuk dari gabungan *graph* ditunjukkan oleh dua *layout* sederhana dari *catering deck* pada kapal penumpang dengan menggunakan kesamaan *topological pattern* seperti ditunjukkan pada gambar dibawah ini.



Gambar 3.3. Contoh dua buah *layout* pada *catering deck* (Manfaat dkk., 1998).



Gambar 3.4. Representasi dalam bentuk *graph* (Manfaat dkk., 1998).



Gambar 3.5. Gabungan *graph* hasil matching dari dua *layout* di atas (Manfaat dkk., 1998).

Pada langkah pertama pembentukan/penyusunan bentuk *graph* adalah menyusun *node graph* dari pasangan-pasangan *node* pada dua *layout* yang memiliki properti (sifat pada ruang) yang sama. Sifat dari dari masing-masing ruangan diungkapkan dari nama ruangan-ruangan tersebut. Sebagai contoh pada dua ruang misalnya *galley* 1 dan *galley* 2 adalah memiliki properti yang sama yaitu *galley*. Kemudian hasil dari langkah pertama adalah *node* (A1 : A2), (B1 : B2) dan seterusnya. Langkah berikutnya adalah menghubungkan dua *node* yang memberikan garis kerja yang sesuai, yang mana memenuhi hubungan kedekatan antara dua ruang pada *layout*. Sebagai contoh, ruang A1 berdekatan dengan ruang B1 pada *layout* 1 dan ruang A2 berdekatan dengan ruang B2 pada *layout* 2. Jadi gabungan *graph* pada *node* (A1 : A2) dapat dihubungkan pada *node* (B1 : B2). Hasil dari gabungan *graph* dapat ditunjukkan pada gambar 3.5.

Dengan gabungan *graph*, kesesuaian terbaik antara dua *graph* pada gambar 3.4 dapat dicari. Pencapaian dari penemuan kumpulan terbanyak dari hubungan

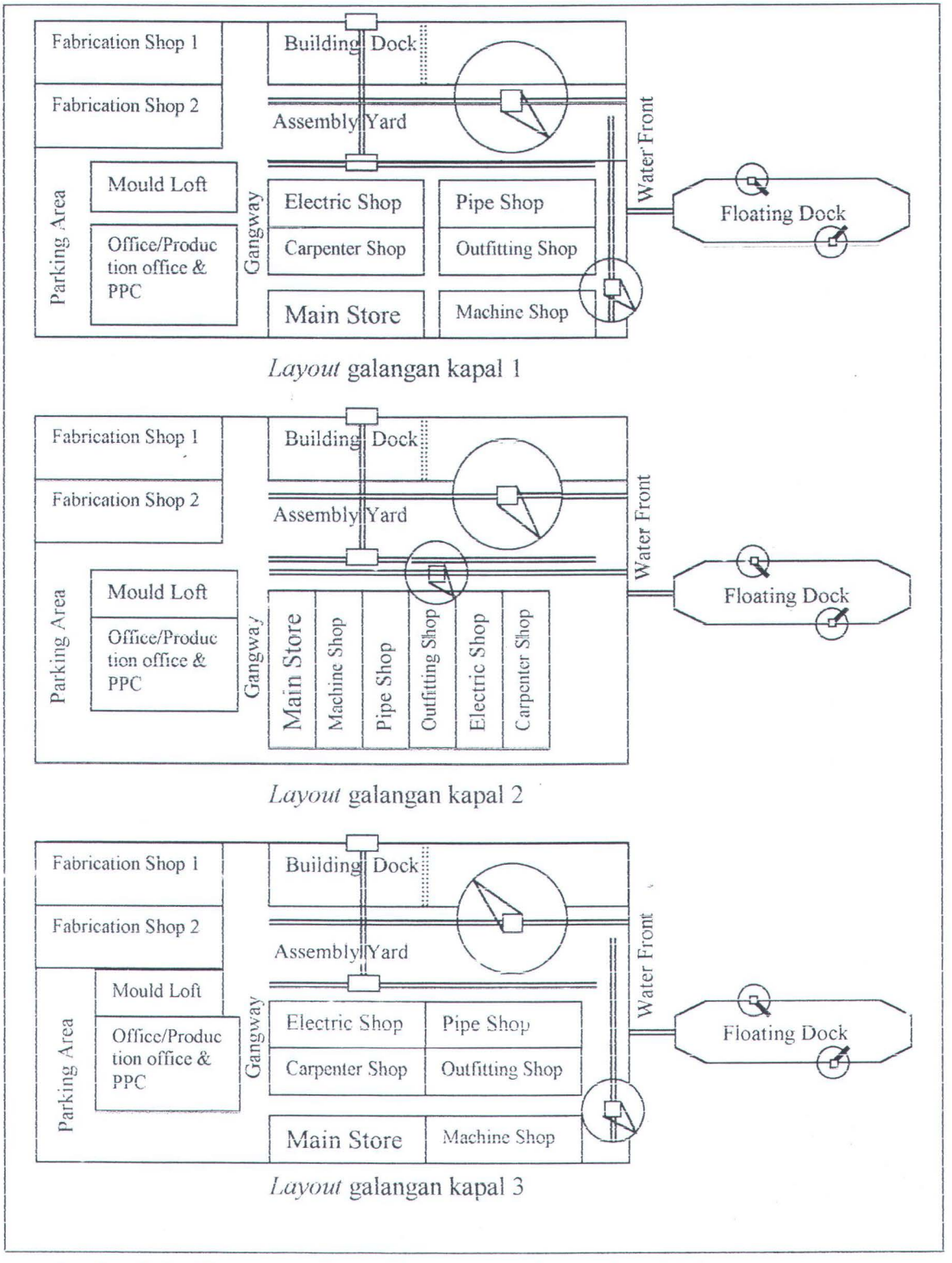
korespondensi *node* pada gabungan *graph* antara satu dengan yang lain memiliki kesesuaian dalam hubungannya, sehingga dapat diperoleh jumlah terbanyak dari kumpulan hubungan dari *node* yang disebut dengan *clique*. Sebagai contoh dalam gabungan *graph* pada gambar 3.5, *clique* terbanyak adalah kumpulan *node* (A1 : A2), (B1 : B2), (D1 : D2) dan (S1 : S2) yang direpresentasikan sebagai kesesuaian terbaik antara dua *layout* pada gambar 3.3.

3.2.1 APLIKASI PADA *LAYOUT* GALANGAN KAPAL

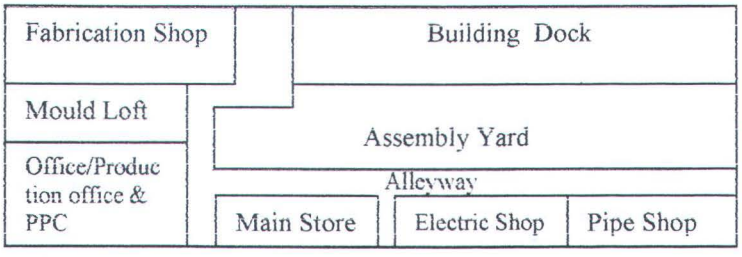
Pendekatan dengan *Topological Pattern (Graph) Matching* yang telah dijelaskan di atas dapat diaplikasikan pada objek *layout* galangan kapal yang diproses pada sistem komputer. Sistem tersebut merupakan implementasi yang digunakan Harlequin LispWorks *(Common Lisp and CLOS)* (The Harlequin Group Limited, 1994). Pada bagian ini kita dapat mempresentasikan dan mendiskusikan secara singkat dalam mencari kesamaan tata letak *(layout)* galangan kapal yang disimpan pada *case base* dalam sistem terhadap *layout template*. *Template* digunakan sebagai dasar dari pengenalan pola yang berarti *input* pola ke dalam *pattern matching process* (Hall and Matias, 1993).

Sebagai contoh terdapat kumpulan tiga macam *layout* galangan kapal sedaerhana yang ditunjukkan pada gambar 3.6 yang telah disimpan pada sistem. Kemudian diberikan sebuah *layout template* (gambar 3.7) sebagai permintaan dari *owner* yang merupakan masukan bagi desainer untuk melakukan proses penyesuaian pola. Hasil kesesuaian dari *template* terhadap *layout* dipresentasikan pada tabel 3.1. Pada tabel ini diurutkan *layout* yang memiliki kesamaan terbanyak

dengan template pada urutan teratas dan *layout* yang memiliki kesamaan terbanyak kedua dan seterusnya pada urutan berikutnya. Urutan dari *layout* tersebut berupa angka hasil hubungan korespondensi dari ruang-ruang. Dalam tabel disebutkan pada layout galangan 3 memiliki angka kesamaan tertinggi yaitu 9, dengan demikian maka *layout* galangan tersebut ditempatkan pada urutan pertama, kemudian diikuti dengan *layout* galangan 2 dengan angka kesamaan 6 dan *layout* galangan 1 dengan angka kesamaan 4.



Gambar 3.6. Tiga macam kumpulan *layout* galangan kapal dalam *case base*



Gambar 3.7. *Layout template*

Layout template	***Layout* galangan kapal**	**Angka kesamaan antar ruang**	**Ruangan-ruangan/spaces yang memiliki hubungan korespondensi**
Layout template	Galangan 3	9	*Fabrication Shop, Mould Loft, Office, Main Store, Electric Shop, Pipe Shop, Assembly Yard, Building Dock dan Gangway*
	Galangan 2	6	*Mould Loft, Office, Main Store, Assembly Yard, Building Dock dan Gangway*
	Galangan 1	4	*Main Store, Assembly Yard, Building Dock dan Gangway*

Tabel 3.1. Hasil kesesuaian antara *layout template* pada gambar 3.7 dan kumpulan *layout* galangan kapal dalam *case base* pada gambar 3.6

Dengan demikian dapat dipilih *layout* galangan yang mendekati dengan keinginan *owner* yaitu *layout* galangan 3 dan selanjutnya oleh desainer, *layout* galangan 3 tersebut dapat dilakukan modifikasi untuk proses penyempurnaan selanjutnya.

BAB IV

SISTEM AISY

BAB IV

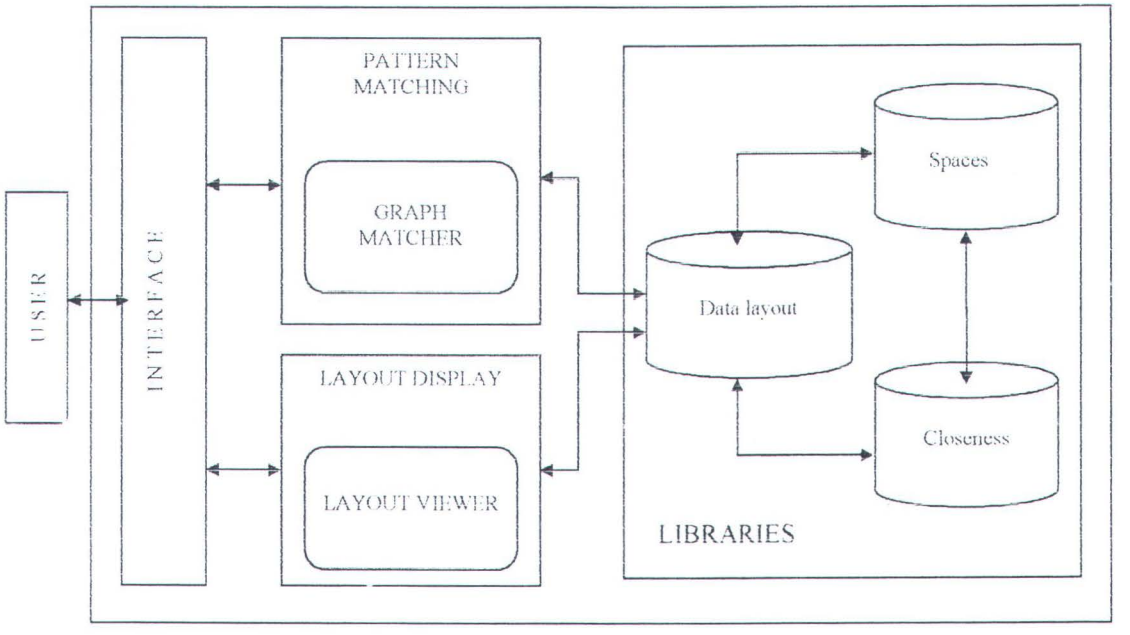
SISTEM AISY

AISY adalah sistem yang mengimplementasikan pendekatan *CBR* dengan teknik *topological pattern (graph) pattern matching* dengan memanfaatkan desain-desain lama untuk proses penyelesaian desain baru. Teknik ini merupakan proses inti dalam *pattern matching* untuk menarik kasus desain tertentu. AISY adalah sebuah alat *(tools)* untuk melakukan *pattern matching layout* dan dapat juga melakukan penarikan *layout* desain dari *case base*. Jika pada sistem diberikan *input* berupa *layout template*, maka sistem akan mengadakan proses pencarian data *layout* galangan kapal yang sudah tersimpan dalam *case base*. Dengan mencari kesesuaian antara *input* dengan data dalam *case base*, akan diperoleh data *layout* yang memiliki angka kesamaan tertinggi (*clique*).

Untuk dapat menjelaskan bagaimana sistem AISY bekerja dalam mendukung pemanfaatan desain lama untuk proses desain baru (sampai dengan tahap penarikan desain) akan dijelaskan pada sub bab berikut ini.

4.1 PEMODELAN SISTEM

Pemodelan sistem AISY digambarkan dalam gambar 4.1 dibawah ini. Sistem ini terdiri dari empat komponen utama, yaitu : *Libraries, layout viewer, pattern matcher* dan *interface*. Sistem ini merupakan implementasi dari Program Harlequin LispWorks (Common Lisp) (The Harlequin Group Limited 1994) dan digunakan pada komputer Pentium II 450 berbasis windows.



Gambar 4.1 Pemodelan sistem AISY.

4.1.1 *LIBRARIES*

Bertujuan untuk menampilkan desain *layout* tertentu, penarikan data *layout* setelah diberikan *input (layout template)*. Data terdiri dari *data layout, spaces* dan *adjacencies. Libraries* dihubungkan secara langsung dengan komponen lain dalam sistem (selain *interface*). *Libraries* adalah komponen dalam *layout cases* yang menghubungkan data *layout* dengan *spaces* dan *closeness* dan dapat diperbaharui dengan memasukkan data-data baru.

Setiap komponen *libraries* akan dijelaskan dalam bagian ini.

Data layout

Isi dari data *layout* berupa kumpulan desain *layout* galangan kapal yang direpresentasikan dengan beberapa attribut seperti nama, area dan daftar *layout.*

Setiap *layout* dalam daftar attribut dari *layout cases* menunjukkan komponen spaces dan kedekatan antar ruang (*closeness*).

Spaces

Berisi kumpulan ruangan pada galangan kapal yang merupakan komponen dari data *layout*. Setiap ruangan adalah *unique*, karena memiliki nama yang berbeda satu sama lain tergantung pada *layout* dimana ruang itu berada.

Closeness

Berisi kumpulan komponen kedekatan antar ruangan dalam data *layout*, hubungan kedekatan antar ruangan adalah *unique*.

4.1.2 *LAYOUT VIEWER*

Merupakan komponen dari sistem AISY yang digunakan untuk menampilkan gambar desain *layout* berdasarkan kebutuhan *user*.

4.1.3 *PATTERN MATCHING*

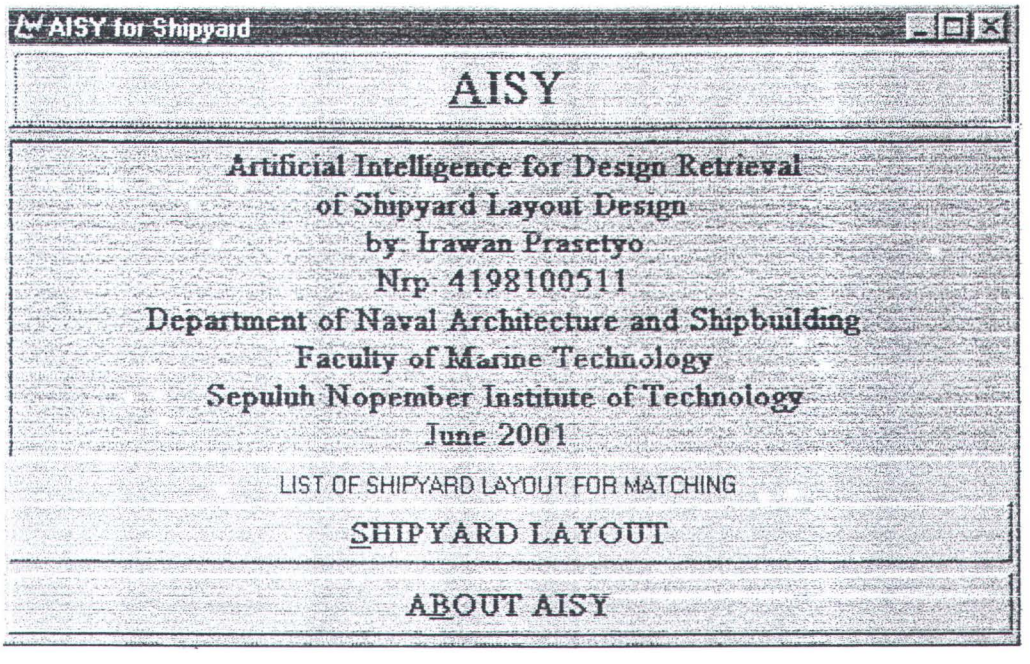
Adalah komponen yang digunakan untuk penarikan desain *layout* setelah diberikan *input* dari desainer. *Graph matcher* digunakan untuk mencari kesesuaian antara *input* dan data dalam *case base* dengan berdasarkan pada *topologi pattern*.

4.1.4 SISTEM *INTERFACE*

Sistem *interface* memberikan fasilitas untuk desainer atau *user* untuk mengoperasikan sistem AISY. Ada tiga bagian yang direpresentasikan dalam

dua *button*:

1. Push *button* untuk *Shipyard Layout*
2. Push *button* untuk *About Aisy*



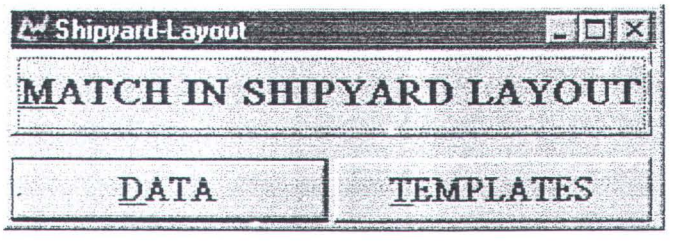
Gambar 4.2 Menu utama sistem AISY.

Tiap-tiap bagian akan dijelaskan dibawah ini :

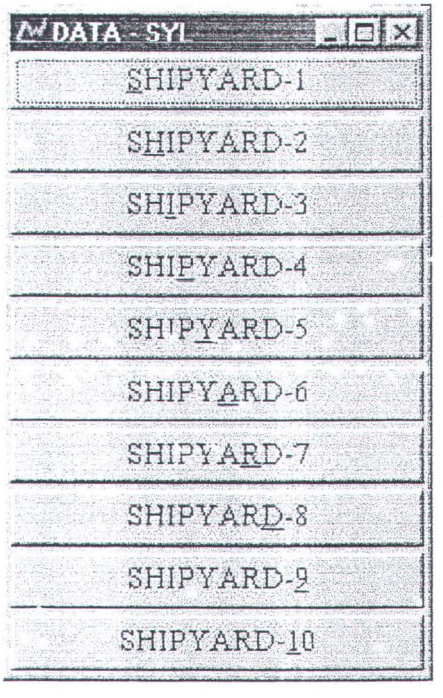
Shipyard Layout

Jika *button shipyard layout* ditekan, sistem akan menampilkan window yang berisi *button* data dan *templates* pada *shipyard layout*. Apabila dipilih *button* data, maka sistem akan menampilkan sepuluh data *layout* galangan kapal yang tersimpan dalam *case base*. *Layout* itu dinamakan Shipyard-1, Shipyard-2,.....Shipyard-10. Sedangkan apabila *button templates* dipilih, sistem akan menampilkan tiga data *template*. *Template* tersebut dinamakan shipyard template-

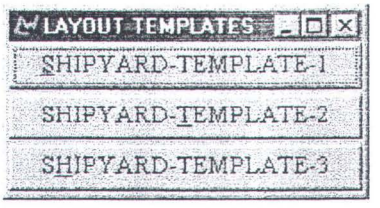
1, shipyard template-2 dan shipyard template-3. *Template* ini merupakan sketsa awal berdasarkan *input* dari *owner* sesuai dengan keinginannya dan disesuaikan dengan aspek-aspek dalam perencanaan *layout* galangan kapal seperti bentuk/tipe galangan, proses produksi, *load faktor*, *material handling* dan sebagainya.



Gambar 4.3 Window untuk Shipyard-Layout

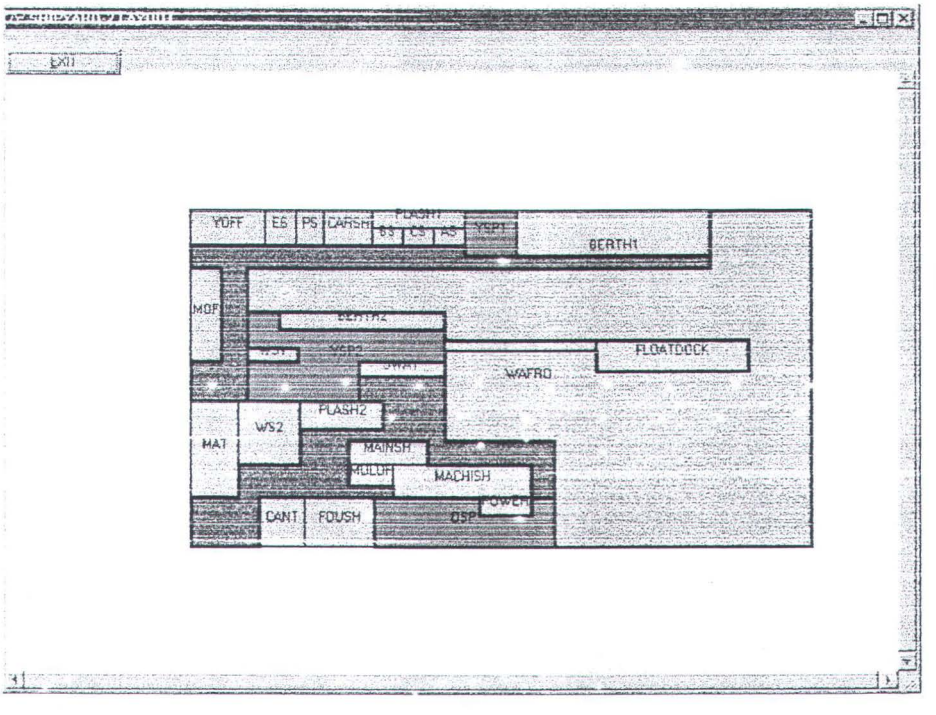


Gambar 4.4 Window untuk Data Shipyard-Layout.

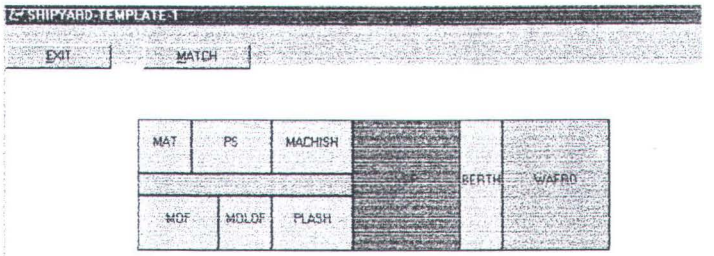


Gambar 4.5 Window untuk Layout Templates

Untuk menampilkan *layout* data, dari sepuluh data yang tersedia dalam window data. Bila *Shipyard-2* dipilih, sistem akan menampilkan window yang berisi *layout* desain galangan kapal 2. Hal ini berlaku pula untuk window *template* dan bila *matching* akan dapat dipilih dilakukan, maka *button* matching pada window *layout template* dipilih.

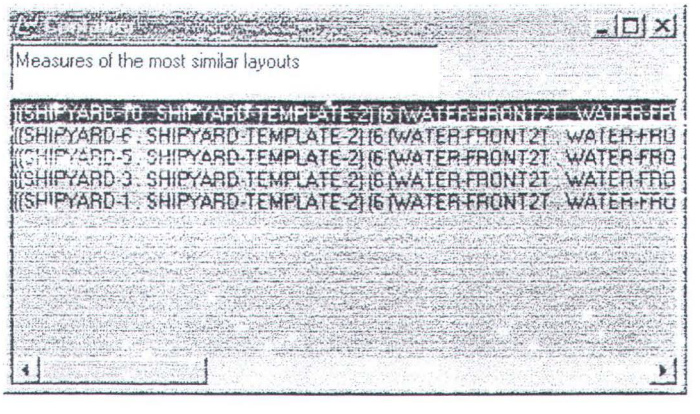


Gambar 4.6 *Window Shipyard Layout-2*

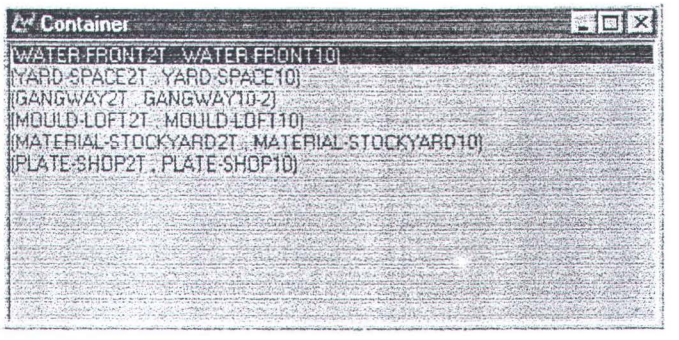


Gambar 4.7 *Window Layout Template-1*

Cara untuk proses *matching* dalam sistem ini terbagi atas dua bagian yaitu hanya dengan mencari *layout* galangan kapal yang memiliki kesamaan tertinggi (*most similarity layout*) atau ingin mengetahui urutan data *layout* dari yang paling tinggi sampai yang terendah berdasarkan angka kesamaan yang dimilikinya. Pada gambar 4.8. terlihat hasil matching antara *layout template-2* dan data *layout* dalam *case base* dengan menggunakan cara pertama. Di sini didapatkan lima buah data *layout* galangan yang memiliki angka kesamaan tertinggi yang sama, sebanyak enam buah.

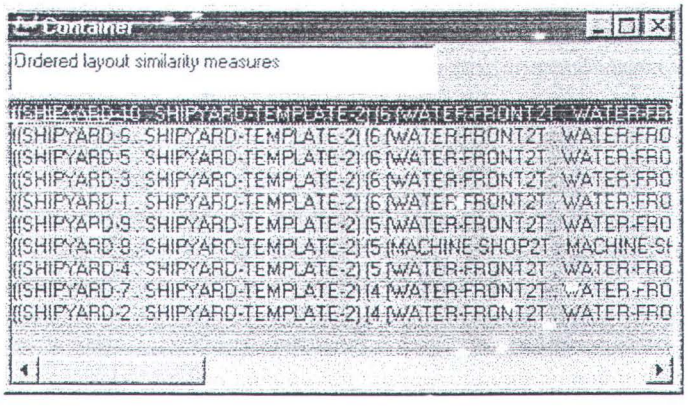


Gambar 4.8 *Most Similarity Layout Template*-2 dengan *Shipyard Layouts*.



Gambar 4.9 Kesamaan hubungan ruangan yang didapat

Hasil proses *matching* secara keseluruhan dapat dilihat dengan menggunakan cara kedua seperti ditunjukkan pada gambar 4.10.



Gambar 4.10 Hasil *matching* antara *Layout Template-2* dengan semua data *layout* dalam *case base*

Seluruh proses diatas berlaku juga untuk proses *matching* antara *layout template-1* dan *layout template-3* dengan data *layout* galangan kapal *(shipyard layouts)* dalam *case base.*

About AISY

Berisi informasi mengenai sistem yang dibuat, pembuat dan spesifikasi lainnya yang berhubungan dengan penggunaan sistem ini.

BAB V

ANALISA HASIL

BAB V

ANALISA HASIL

Dari hasil kesesuaian (*matching result*) yang telah dijalankan sistem AISY antara tiga buah *layout template* dengan data *layout* galangan *kapal (shipyard layout)* didapatkan tiga buah tabel *matching result* untuk masing-masing *layout template.*

Masing-masing tabel tersebut berisi :

- Angka kesamaan *(clique)* antara *layout template* dengan masing-masing *layout* galangan kapal.
- Urutan data *layout* galangan kapal berdasarkan angka kesamaan hasil proses matching yang disusun berurutan dari *clique* tertinggi hingga terendah.
- Nama ruangan *(space)* hasil proses *matching* yang memiliki hubungan korespondensi

Data *layout* galangan kapal selengkapnya dapat dilihat pada bagian lampiran, sedangkan tabel *matching result* masing-masing *template* ditunjukkan oleh tabel 5.1., tabel 5.2. dan tabel 5.3. dibawah ini. Data *layout* tersebut berjumlah sepuluh buah terdiri dari *Shipyard-1* sampai dengan *Shipyard-10*, sedangkan *layout template* berjumlah tiga buah terdiri *dari layout tempate-1* sampai dengan *layout template-3.*

Tabel 5.1 *Matching Result* antara *Layout Template-1* dengan data *layout* galangan kapal *(Shipyard Layouts)*

Layout Templates	Clique	Shipyard Layouts	The Corresponding Space
Layout Template-1	7	Shipyard Layout-6	Water Front, Building Berth, Plate Shop, Main Office, Machine Shop, Piping Shop dan Material Stockyard
	6	Shipyard Layout-9	Water Front, Building Berth, Plate Shop, Main Office, Machine Shop dan Material Stockyard
	6	Shipyard Layout-5	Water Front, Building Berth, Plate Shop, Main Office, Machine Shop dan Material Stockyard
	6	Shipyard Layout-4	Water Front, Building Berth, Plate Shop, Main Office, Machine Shop dan Material Stockyard
	6	Shipyard Layout-2	Water Front, Building Berth, Plate Shop, Main Office, Machine Shop dan Material Stockyard
	6	Shipyard Layout-1	Water Front, Building Berth, Plate Shop, Main Office, Machine Shop dan Piping Shop
	5	Shipyard Layout-10	Water Front, Building Berth, Gangway, Main Office dan Material Stockyard
	5	Shipyard Layout-7	Water Front, Building Berth, Plate Shop, Main Office dan Machine Shop
	5	Shipyard Layout-3	Water Front, Gangway, Plate Shop, Main Office dan Piping Shop
	4	Shipyard Layout-8	Water Front, Gangway, Mould Loft dan Material Stockyard

Analisa hasil dari ketiga tabel tersebut antara lain :

1. Hasil proses *maching* berupa *clique* pada ketiga tabel berbeda satu sama lainnya. Hal ini dikarenakan masing-masing *layout template* memiliki jumlah ruangan yang berbeda, nama ruangan yang berbeda dan kedekatan antar ruang yang bebeda.
2. *Layout Template* sebagai *input* dari desainer bukan sebagai *layout* yang akan dimodifikasi karena sesuai dengan model CBR, yang akan dimodifikasi adalah

Tabel 5.2 *Matching Result* antara *Layout Template-2* dengan data *layout* galangan kapal *(Shipyard Layouts)*

Layout Templates	Clique	Shipyard Layouts	The Corresponding Space
Layout Template-2	6	Shipyard Layout-10	Water Front, Yard Space, Gangway, Mould Loft, Material Stockyard dan Plate Shop
	6	Shipyard Layout-6	Water Front, Machine Shop, Main Office, Parking Area, Mould Loft dan Plate Shop
	6	Shipyard Layout-5	Water Front, Machine Shop, Parking Area, Gangway, Mould Loft dan Plate Shop
	6	Shipyard Layout-3	Water Front, Main Office, Parking Area, Gangway, Material Stockyard dan Plate Shop
	6	Shipyard Layout-1	Water Front, Main Office, Parking Area, Gangway, Mould Loft dan Plate Shop
	5	Shipyard Layout-9	Water Front, Machine Shop Main Office, Mould Loft dan Piping Shop
	5	Shipyard Layout-8	Machine Shop, Main Office, Parking Area, Material Stockyard dan Plate Shop
	5	Shipyard Layout-4	Water Front, Main Office, Gangway, Mould Loft dan Plate Shop
	4	Shipyard Layout-7	Water Front, Machine Shop, Gangway dan Plate Shop
	4	Shipyard Layout-2	Water Front, Main Office, Mould Loft dan Plate Shop

data *layout* hasil penarikan dari proses *pattern matching*.

3. Pada tabel 5.1, *Shipyard Layout-6* merupakan *layout* dengan angka kesamaan tertinggi, diikuti oleh *Shipyard Layout-9, 8,* dan seterusnya. Dengan hasil ini, tidak mutlak *Shipyard Layout-6* yang akan dipilih oleh desainer untuk proses adaptasi meskipun memiliki angka kesamaan tertinggi, karena tentunya desainer akan mempertimbangkan atau meninjau kembali dari segi aspek

Tabel 5.3 *Matching Result* antara *Layout Template-3* dengan data *layout* galangan kapal *(shipyard layouts)*

Layout Templates	Clique	Shipyard Layouts	The Corresponding Space
Layout Template-3	7	Shipyard Layout-6	Open Space, Machine Shop, Main Office, Building Berth, Yard Space, Plate Shop and Yard Office
	7	Shipyard Layout-2	Water Front, Machine Shop, Mould Loft, Main Office, Building Berth, Plate Shop and Yard Space
	6	Shipyard Layout-10	Water Front, Open Space, Main Office, Gangway, Building Berth and Plate Shop
	6	Shipyard Layout-9	Mould Loft, Material Stockyard, Gangway, Building Berth, Plate Shop and Yard Office
	6	Shipyard Layout-5	Water Front, Mould Loft, Main Office, Building Berth, Plate Shop and Yard Office
	6	Shipyard Layout-4	Water Front, Machine Shop, Material Stockyard, Gangway, Building Berth and Plate Shop
	6	Shipyard Layout-1	Water Front, Open Space, Mould Loft, Gangway, Plate Shop and Yard Office
	5	Shipyard Layout-8	Water Front, Open Space, Mould Loft, Main Office and Plate Shop
	5	Shipyard Layout-7	Water Front, Machine Shop, Main Office, Building Berth and Plate Shop
	5	Shipyard Layout-3	Water Front, Machine Shop, Gangway, Plate Shop and Yard Office

dalam perencanaan *layout* galangan kapal seperti bentuk/tipe galangan, aliran material, urutan proses produksi dan sebagainya. Kemungkinan dipilihnya *Shipyard Layout-6* adalah teerbesar, tapi tidak menutup kemungkinan *Shipyard Layout-9* atau *5* bahkan mungkin *4* akan dipilih jika ternyata lebih memenuhi dari segi aspek di atas.

4. Hasil penarikan *layout* dengan angka kesamaan tertinggi ini merupakan hasil optimisasi yang mana data *layout* adalah data yang dianggap sudah optimum.

5. Dengan adanya perbedaan hasil kesesuaian antara *Layout Template-1, 2* dan *3* dengan data *layout* galangan kapal menunjukkan sistem AISY telah teruji sehingga sistem ini telah memiliki kemampuan *(capability)* dan kesesuaian *(compatibility)* untuk melakukan proses *pattern matching.*

BAB VI

PENUTUP

BAB VI

PENUTUP

6.1 KESIMPULAN

Dari analisa hasil yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut :

1. Hasil matching menunjukkan adanya konsistensi dari teknik *topological pattern (graph) matching* dalam sistem AISY untuk melakukan simulasi *pattern matching* yang dilakukan oleh manusia. Dengan adanya simulasi komputer tersebut, waktu yang diperlukan untuk proses *matching* menjadi lebih singkat dan obyektif, sehingga jumlah data dapat ditambah untuk mencapai hasil yang diinginkan.
2. Integrasi *CBR* dengan teknik *topological pattern matching* cocok untuk digunakan dalam perencanaan *layout* galangan kapal. Hal itu dapat dilihat dari perbandingan antara proses *pattern matching* secara manual dan dengan menggunakan program komputer.
3. Berdasarkan hasil yang telah dicapai maka tujuan utama dari penulisan tugas akhir telah tercapai, yaitu memperbaiki proses desain tata letak galangan kapal dengan teknik *topological pattern (graph) matching.*
4. Dengan menggunakan aplikasi *CBR* dengan teknik *topological pattern (graph) matching* pada sebuah program komputer (AISY) dapat mengurangi pekerjaan desainer karena waktu perancangan menjadi lebih pendek dan desain yang dihasilkan lebih mendekati keinginan pemesan.

6.2 SARAN

Beberapa saran yang dapat penulis sampaikan berdasarkan hasil yang telah dicapai adalah :

- Untuk memaksimalkan sistem AISY perlu diperhatikan jumlah data yang digunakan, semakin banyak jumlah data akan semakin baik dan variatif.
- Sistem ini diharapkan pada masa yang akan datang dapat melanjutkan proses adaptasi desain *layout* galangan kapal.
- Untuk penggunaan data *layout* galangan kapal yang memiliki jumlah data lebih dari sepuluh buah dan penggunaan data *template* yang memiliki jumlah ruang (*space*) lebih dari sepuluh buah disarankan untuk menggunakan komputer setara Pentium III.

DAFTAR PUSTAKA

DAFTAR PUSTAKA

1. Andjar Soeharto dan Soejitno (1996), **Diktat Galangan Kapal**, Penerbit ITS Surabaya.
2. Andrews, D (1985), *An integrated approach to ship synthesis*, **Transactions of The Royal Institution of Naval Architects** 127, 73-102.
3. Apple, James M. (1990), **Tata letak Pabrik dan Pemindahan Bahan**, Edisi Ketiga , Penerbit ITB Bandung.
4. Domeshek, E. And J. Koldner (1992**), A case-based design aid for architecture.** In J. S. Gero (Ed.), *Artificial Intelligence in Design '92*, Dordrecht, pp. 497-516, Kluwer Academic Publishers.
5. Faltings, B. (1997), **Case Reuse by Model-Based Interpretation**. In M. L. Maher and P. Pu (Eds.), *Issues and Applications of Case-Based Reasoning in Design*, Hills-dale, NJ. : Lawrence Erlbaum Associates.
6. Flemming, U., Z. Aygen, R. Coyne and J. Snyder (1997), **Case-based design in asoftware environment that supports the early phases**. In M. L. Maher and P. Pu (Eds.), *Issues and Applications of Case-Based Reasoning in Design*, Hillsdale, NJ. : Lawrence Erlbaum Associates.
7. Francis, Richard L. (1992), **Facility Layout and Location an Analytical Approch**, Second Edition, Prentice Hall, New Jersey.
8. Foz, A. (1993), *Observations on design behaviour in the parti.* **In Proceedings of the Design Activity Conference.**

9. Guy L. Steele Jr. (March 1984), **Common Lisp** : *The Language*, Pittsburgh, Pennyslvania.

10. Giretti, A., L. Spalazzi and M. Lemma (1994**), A.S.A. An interactive assistant to architectural design.** In J.S. Gero and F. Sudweeks (Eds.*)*, *Artificial Intelligence in Design '94*, The Netherlands, pp.93-108, Kluwer Academic Publishers.

11. Hua, K., I. Smith, B. Faltings, S. Shih and G. Schmitt (1992*)*, ***Adaptation of spatial spatial design cases.*** In J. Gero (Ed.), *Artificial Intelligence in Design, Netherlands*, pp. 559-575. Kluwer Academic Publishers.

12. Hua, K., B. Faltings, and I. Smith (1996), *CADRE : Case-based geometric design.* **Artificial Intelligence in Engineering** 10(2), 171-183.

13. John S. Gero & Fay Sudweeks (1994), **Artificial Intelligence in Design.**

14. Koldner, J.L. (1993), **Case Based Reasoning**, Morgan Kauvmann, New York.

15. Mac Callum, K.J. A.H.B. Duffy, I.A. Donaldson, S.M. Duffy and F. O'Donnell (1997), **Design Reuse- Design Concepts in New Engineering Contexts**, Technical Report, CAD Centre, University of Strathclyde, Glasgow.

16. Maher, M.L. and D.M. Zhang (1991), **Cadsyn : using case and decomposition knowledge for design synthesis**. In J. Gero (Ed.*)*, *Artificial Intelligence in Design, Netherlands*, pp. 137-150. Kluwer Academic Publishers.

17. Maher, M.L., M.B. Balachandran and D.M. Zhang (1995), **Cased-based Reasoning in Design**. Mahwah, New Jersey : Lawrence Erlbaum Associates, Publishers.

18. Maher, M.L. and A. G. D. S. Garza (1997), **Case-based reasoning in design.** IEEE Expert. *Special Issue on Artificial Intelligent in Design.*

19. Manfaat, D., A.H.B. Duffy, and B.S. Lee (1996), **Review of pattern matching approaches,** *The Knowledge Engineering Review* 11 (2), 161-189.

20. Manfaat, D. (April 1998), **Computer-based Approach to the Effective Utilisation of Spatial Layout Design Experience**.

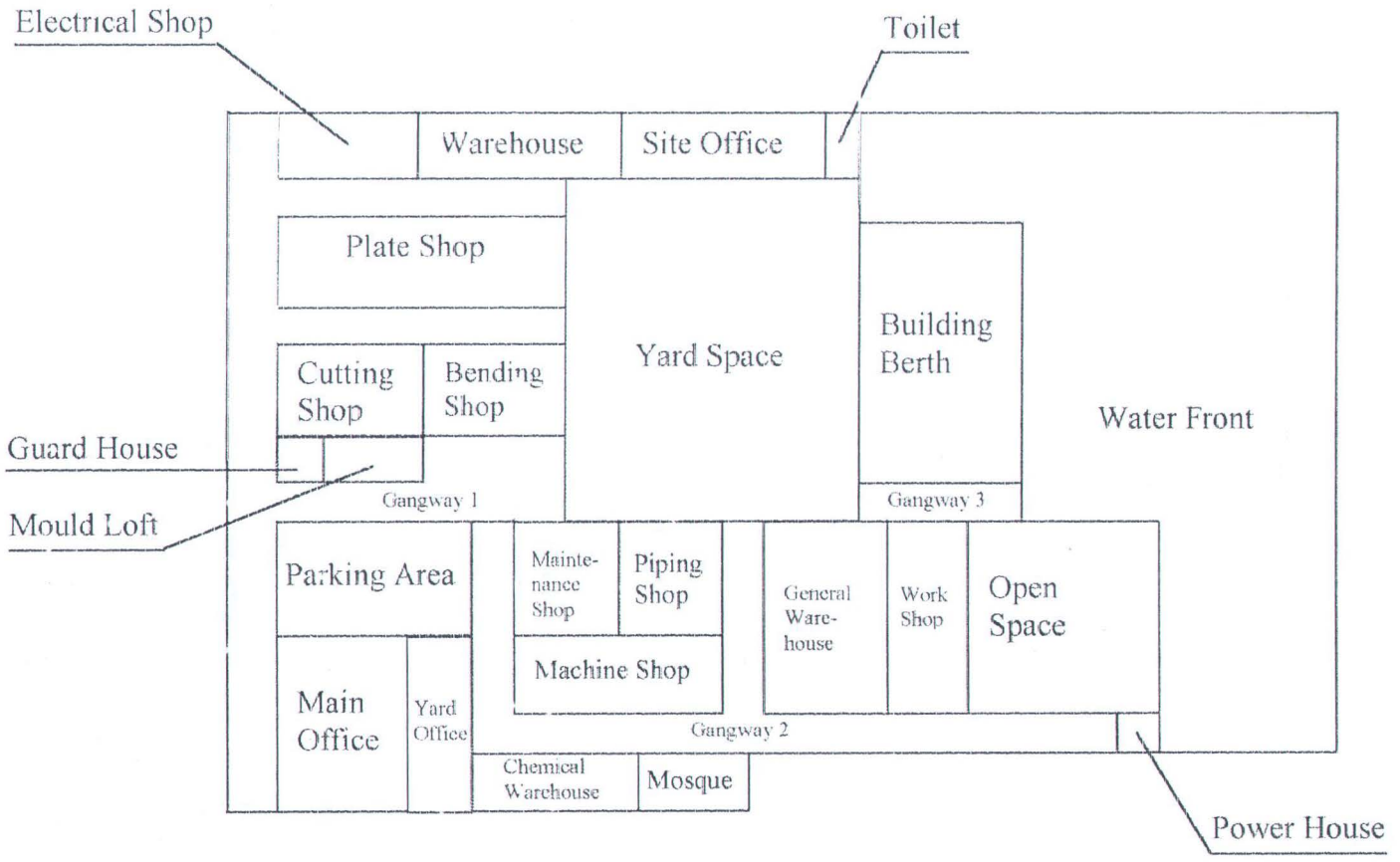
21. Manfaat, D., A.H.B. Duffy, and B.S. Lee (July 1998), **Topological Patern and in Geometric Shape Matching**, *Proceedings of the 1998 Artificial Intelegence Design Conference (AID'98)*, Lisbon, Portugal, p-p. 327-344.

22. Oxman, R. and R. Oxman (April 1992), **Refinement and Adaptation in Design Cognition.** Design Studies 13(2), 117-134.

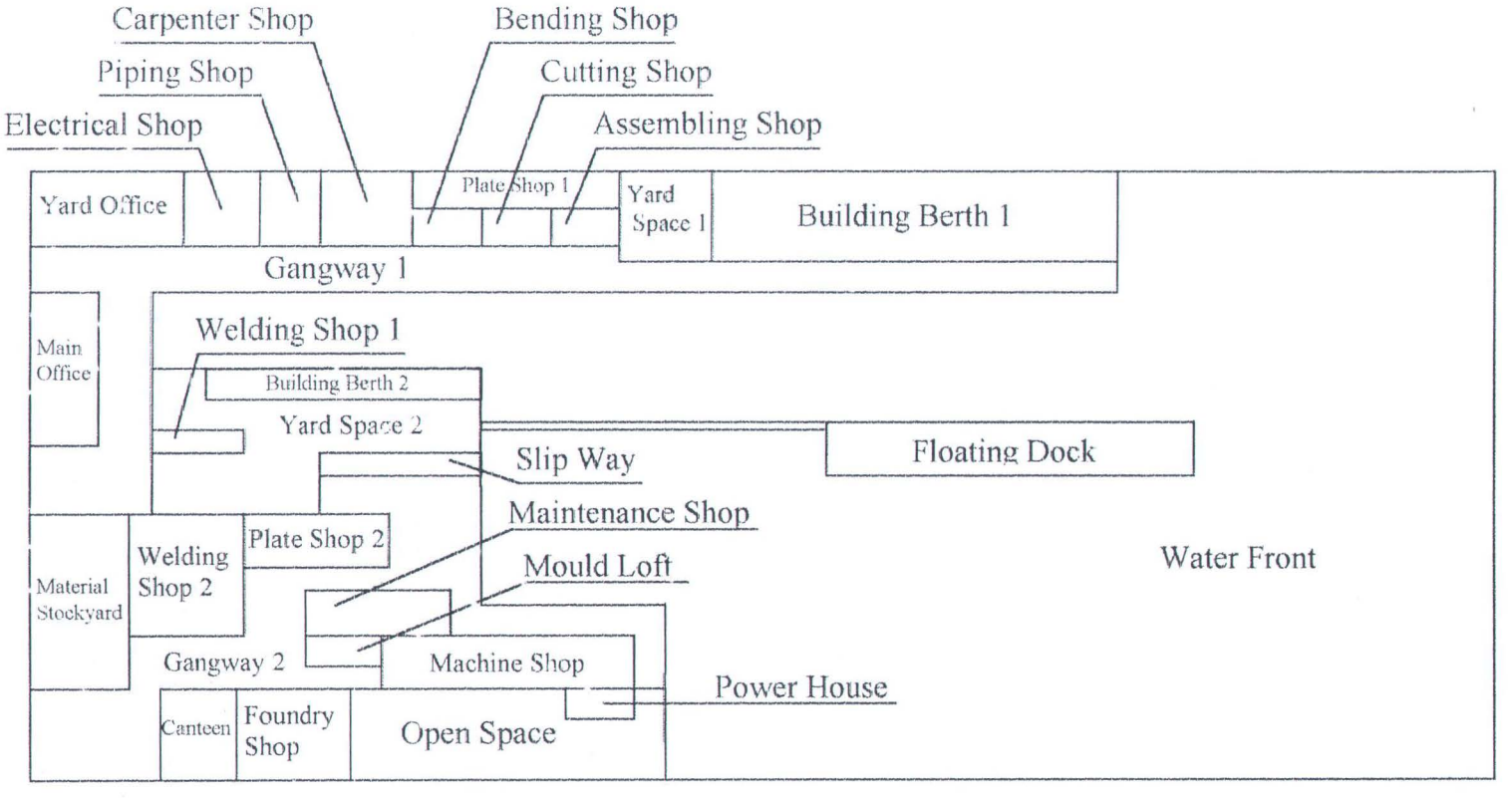
23. Voβ, A., C. H. Coulon, W. Grather, B. Linowski, J.Schaaf, B. Bartsch-Sporl, K. Borner, E. C. Tammer, H. Durschke and M. Knauff (1994), **Retrival of similiar layouts-about avery hybrid approach in FABEL**. In J. S. Gero and F. Sudweeks (Eds.), *Artificial Intelligence in Design '94*, The Netherlands, pp.625-640. Kluwer Academic Publishers

LAMPIRAN

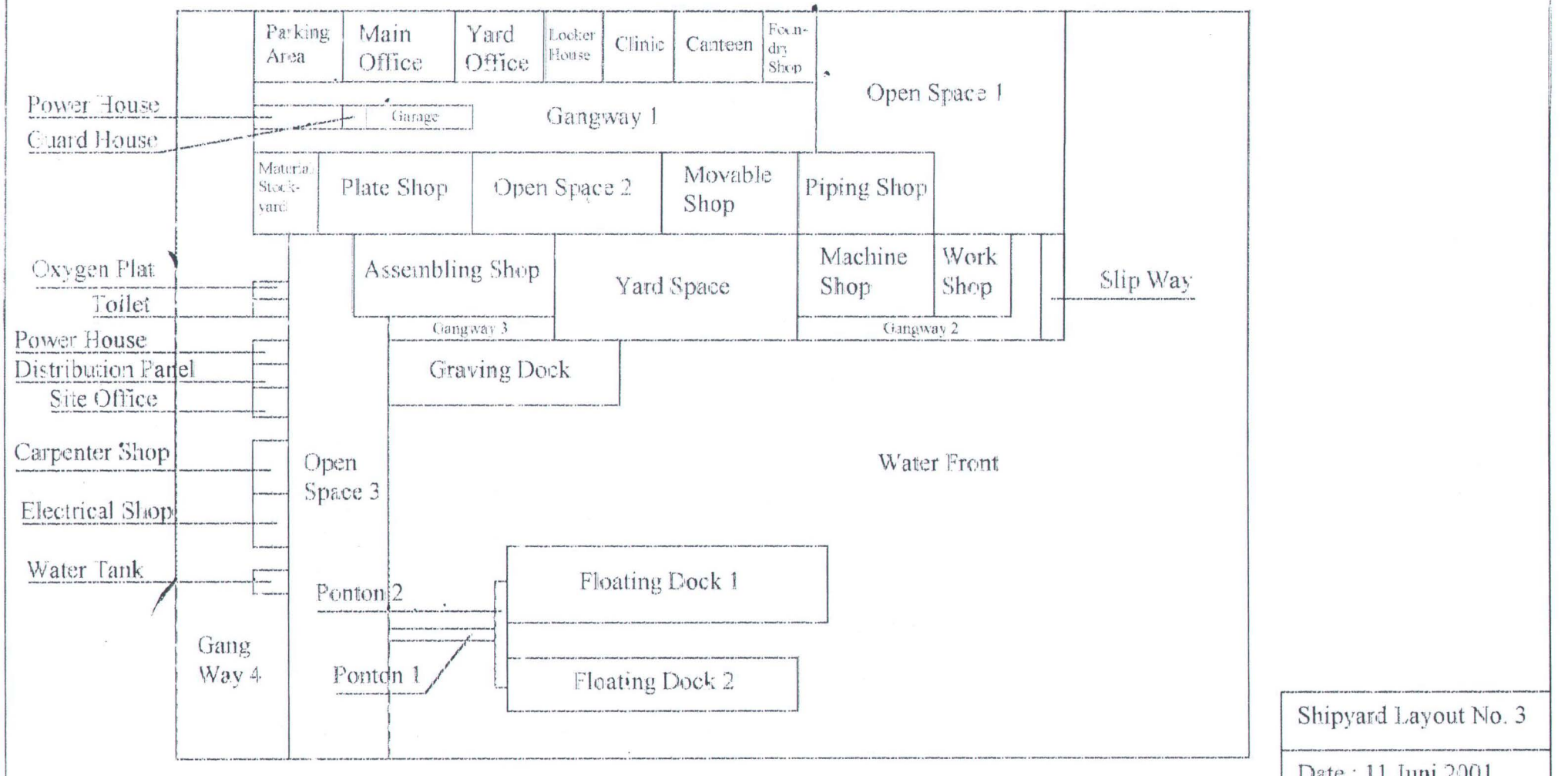
DATA *LAYOUT*



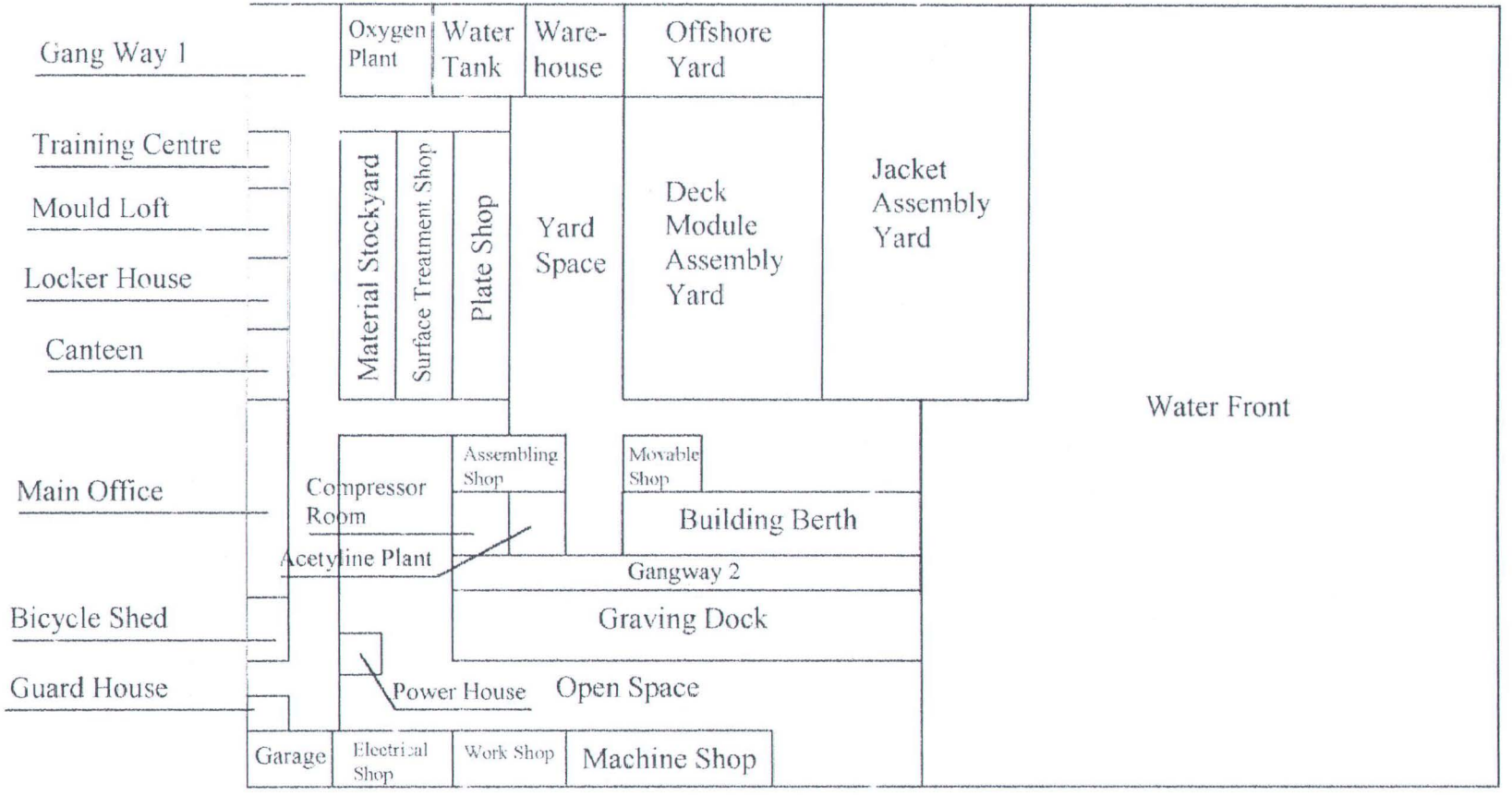
Shipyard Layout No. 1
Date : 11 Juni 2001
Irawan Prasetyo



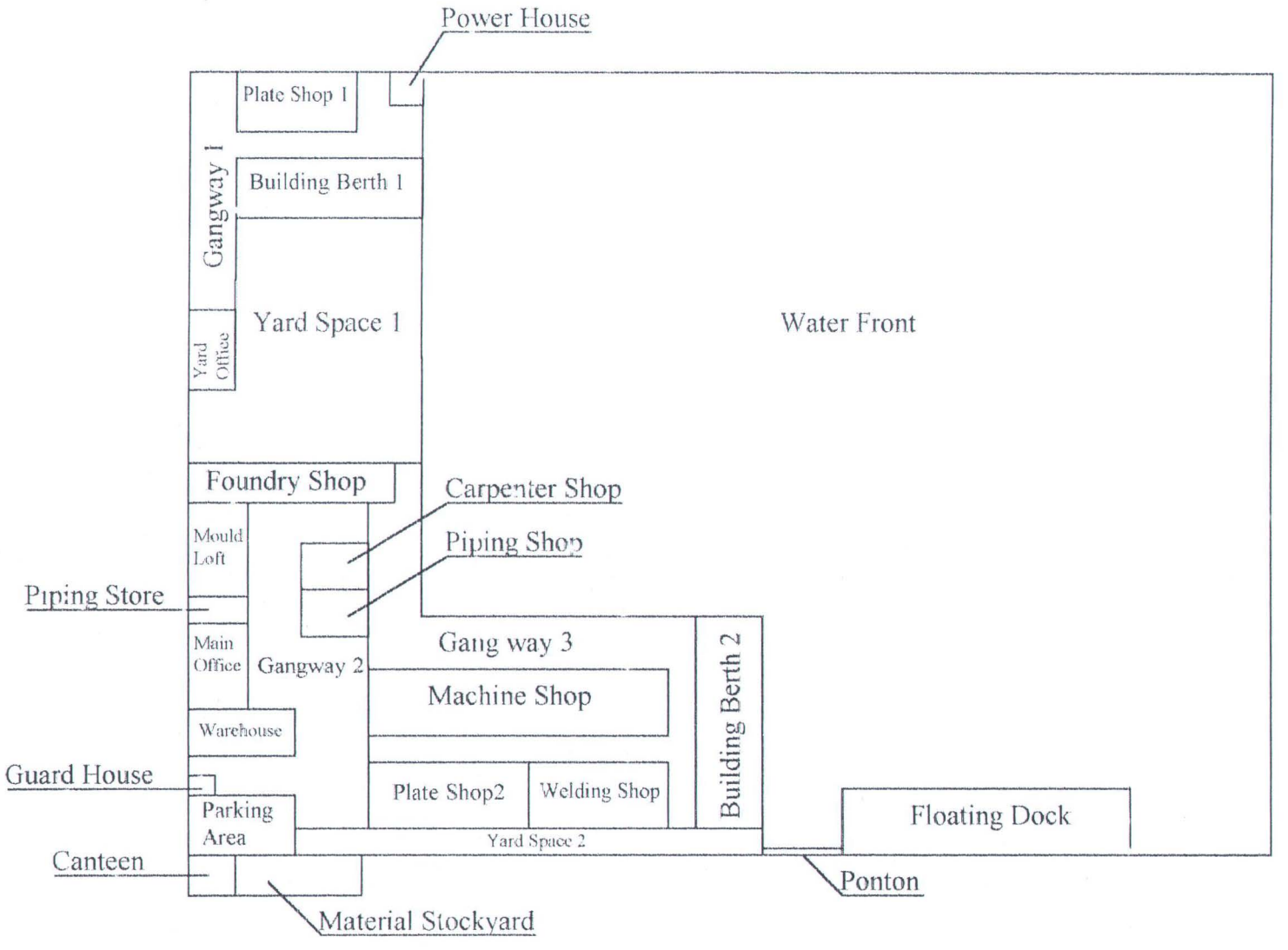
Shipyard Layout No. 2
Date : 11 Juni 2001
Irawan Prasetyo



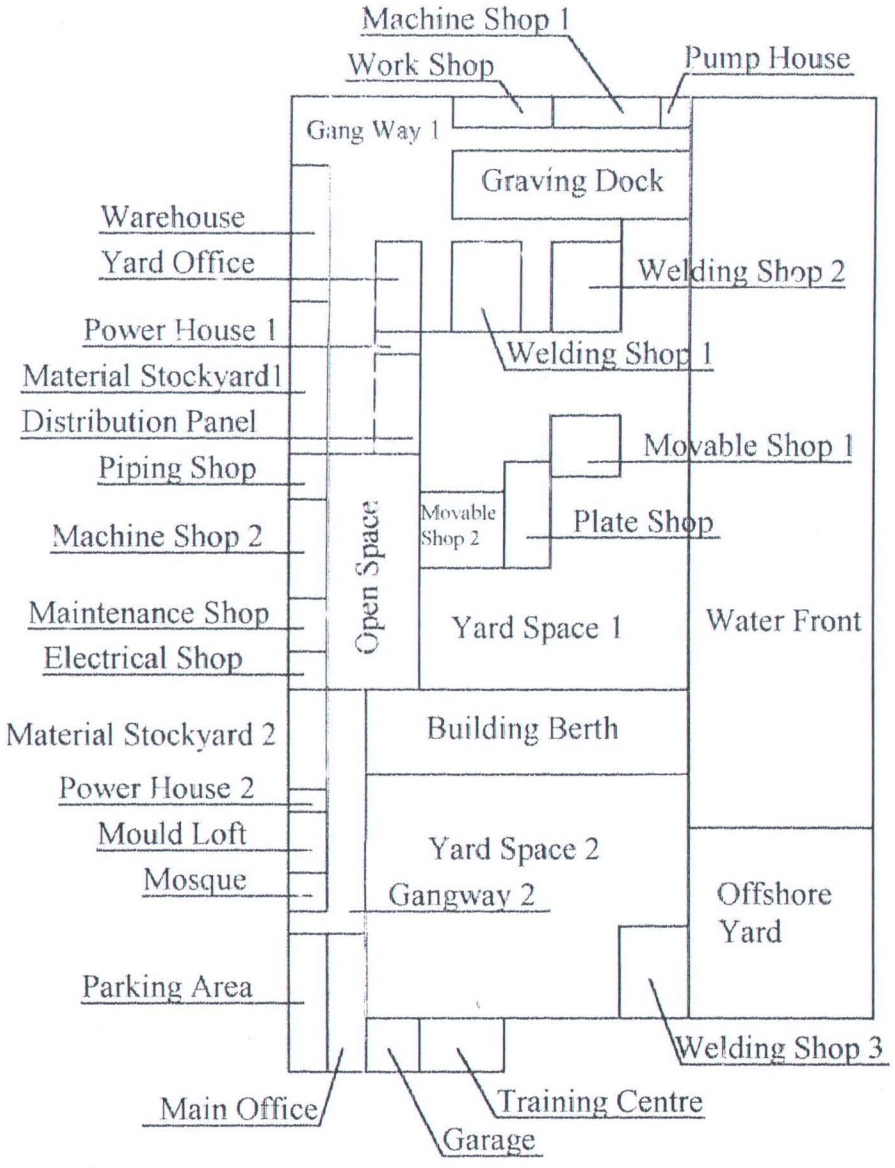
Shipyard Layout No. 3
Date : 11 Juni 2001
Irawan Prasetyo



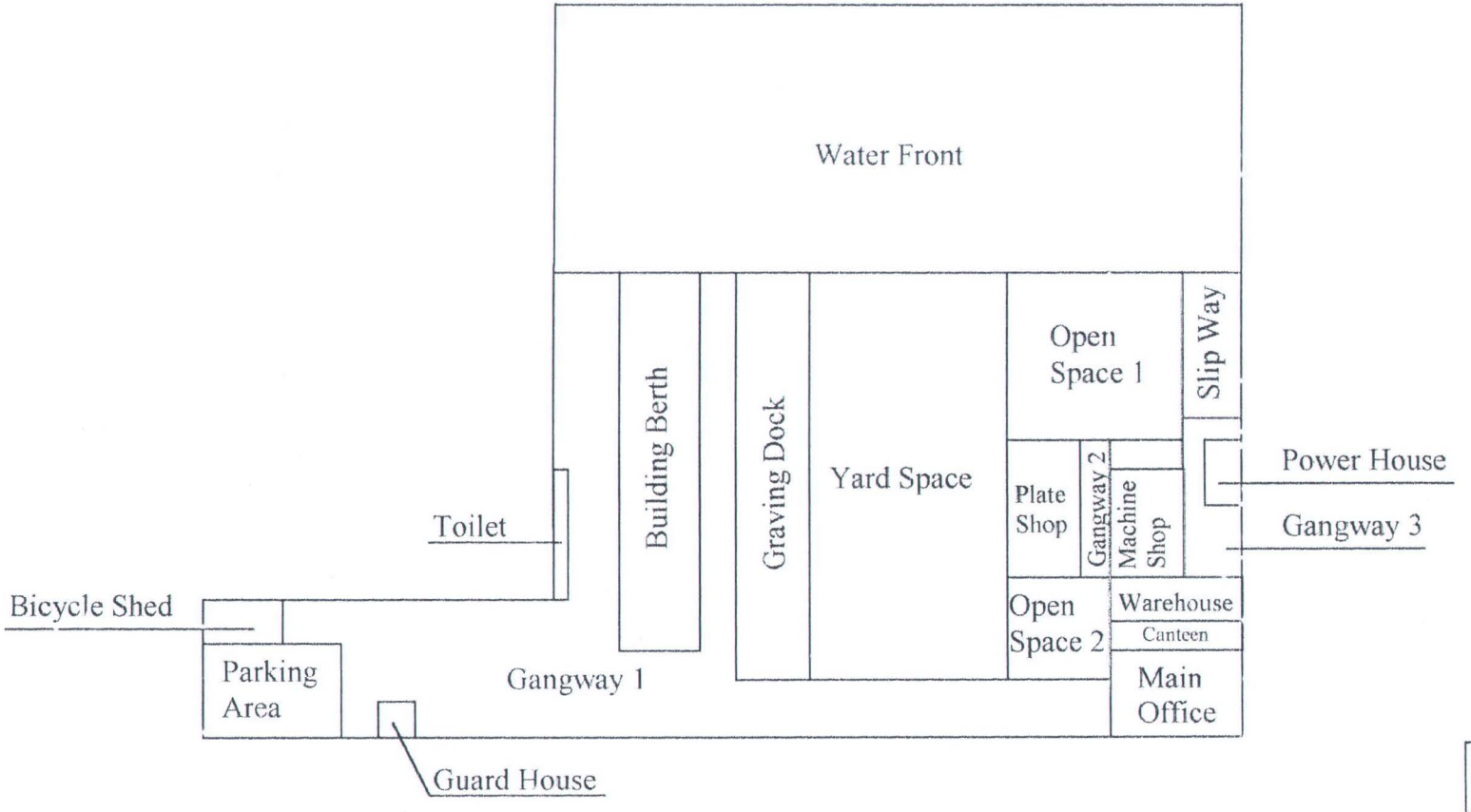
Shipyard Layout No. 4
Date : 11 Juni 2001
Irawan Prasetyo



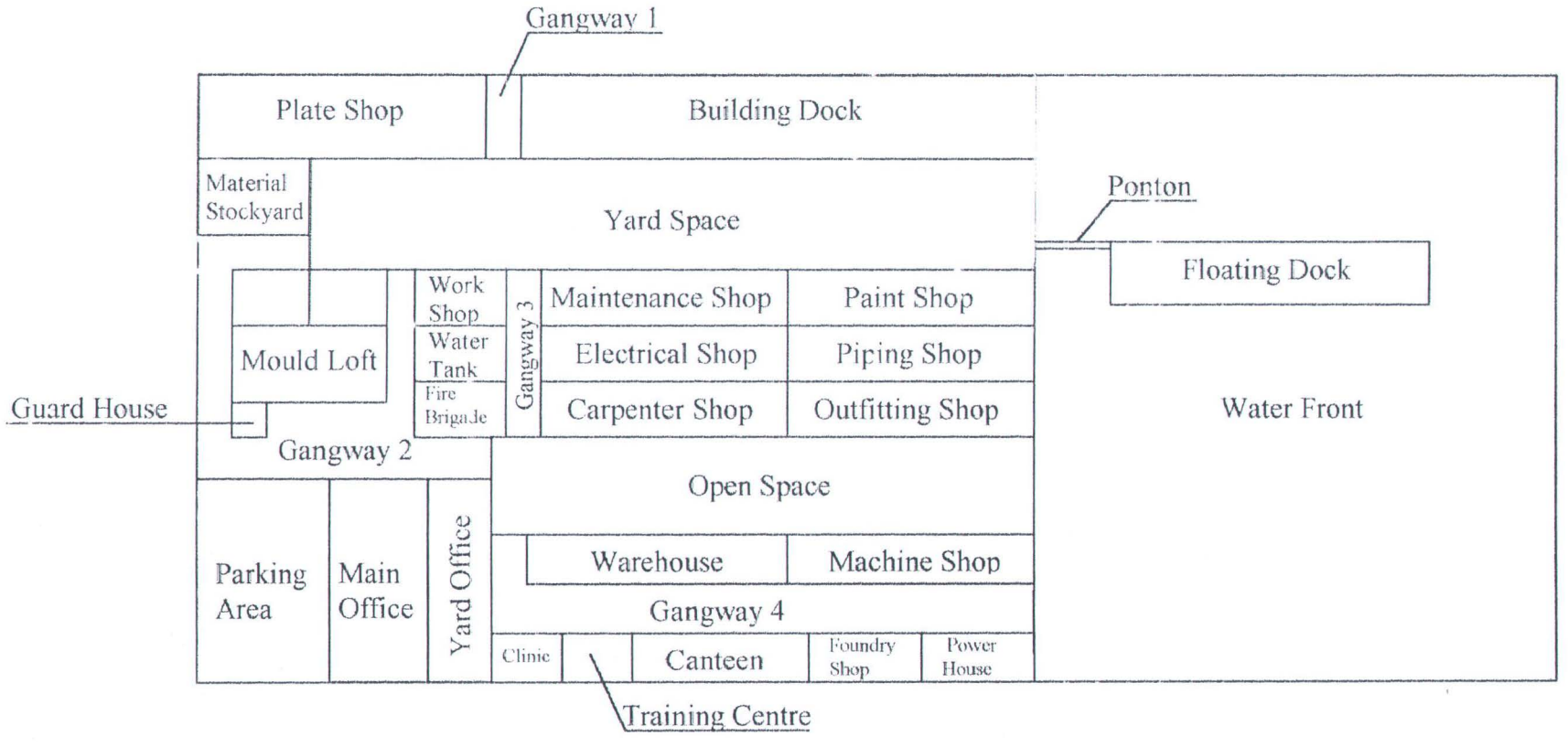
Shipyard Layout No. 5
Date : 11 Juni 2001
Irawan Prasetyo



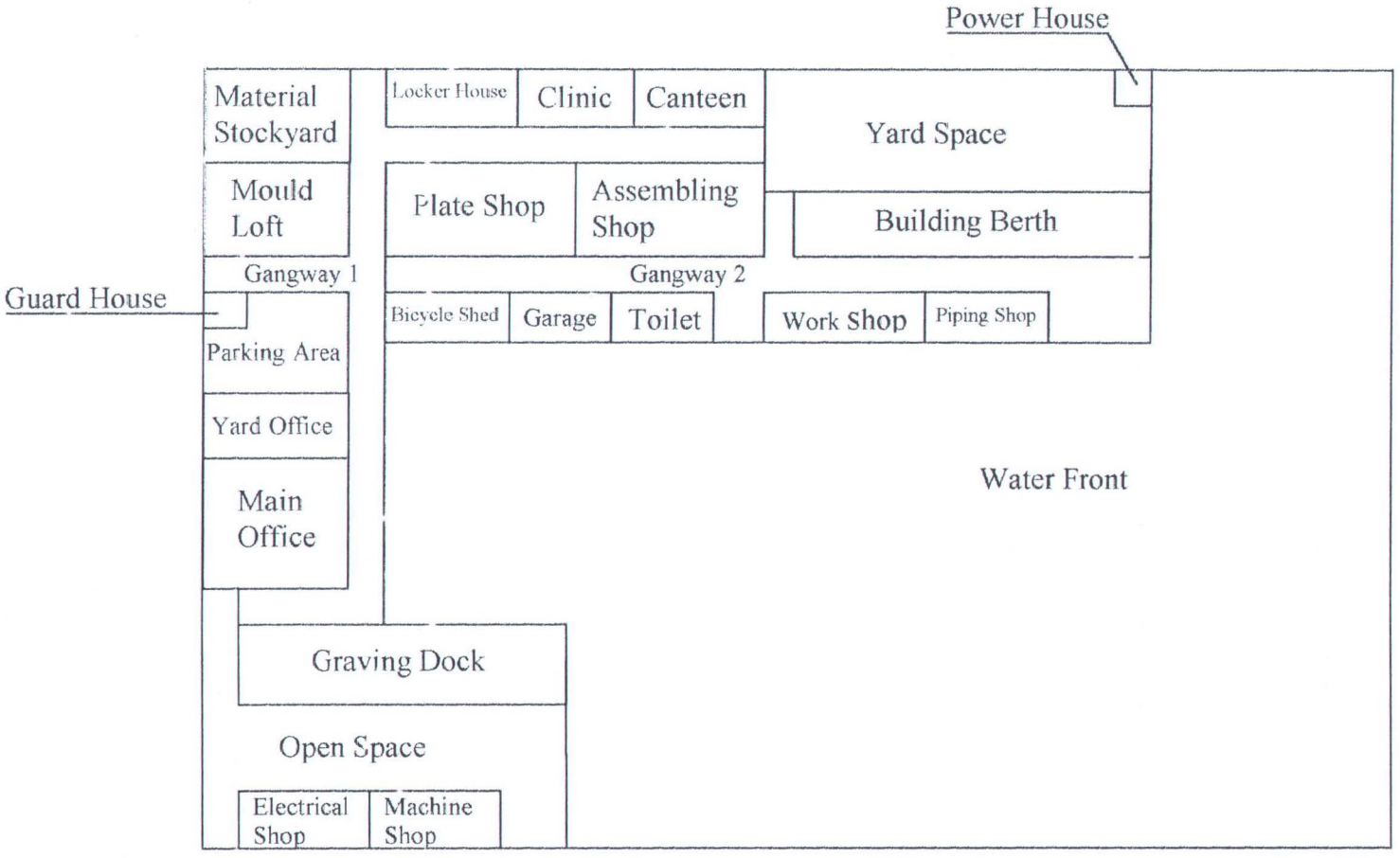
Shipyard Layout No. 6
Date : 11 Juni 2001
Irawan Prasetyo



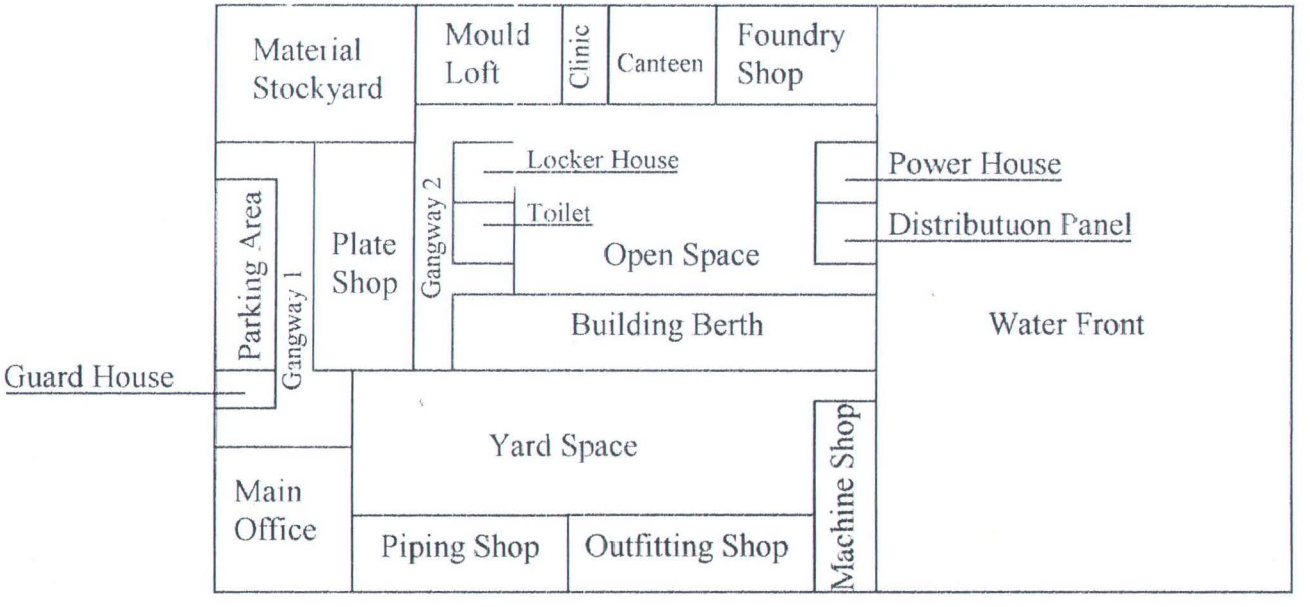
Shipyard Layout No. 7
Date : 11 Juni 2001
Irawan Prasetyo



Shipyard Layout No. 8
Date : 11 Juni 2001
Irawan Prasetyo

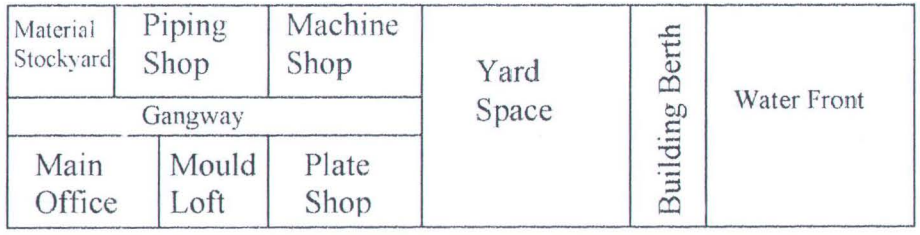


Shipyard Layout No. 9
Date : 11 Juni 2001
Irawan Prasetyo

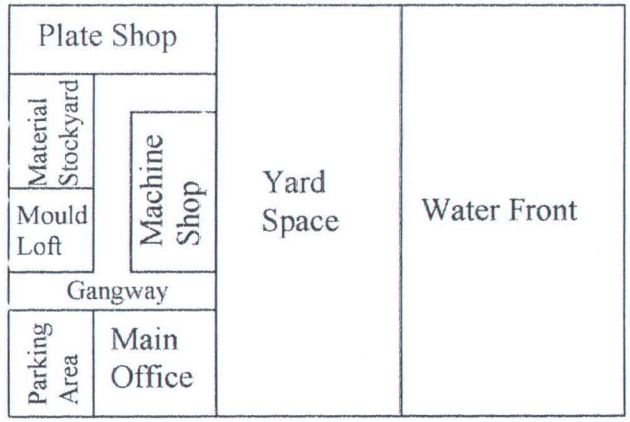


Shipyard Layout No.10
Date : 11 Juni 2001
Irawan Prasetyo

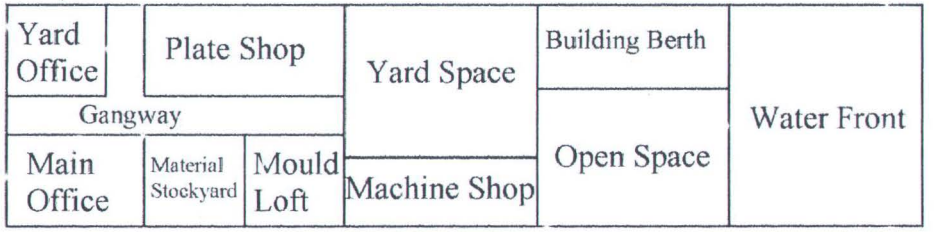
DATA *TEMPLATE*



Layout Template No. 1
Date : 11 Juni 2001
Irawan Prasetyo

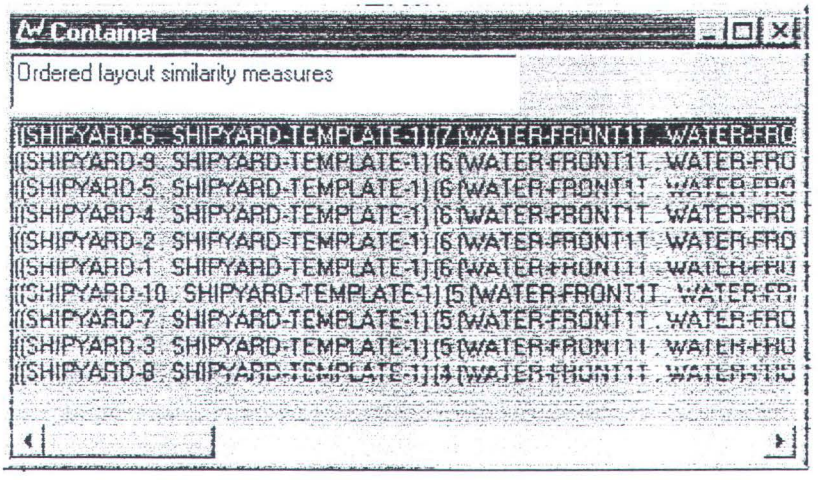


Layout Template No. 2
Date : 11 Juni 2001
Irawan Prasetyo

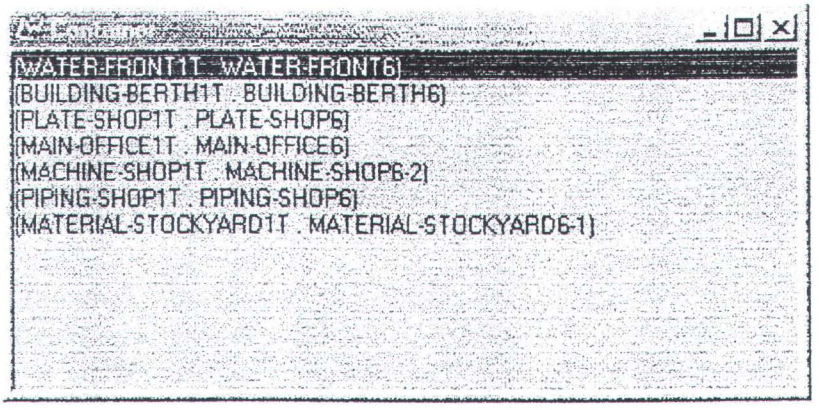


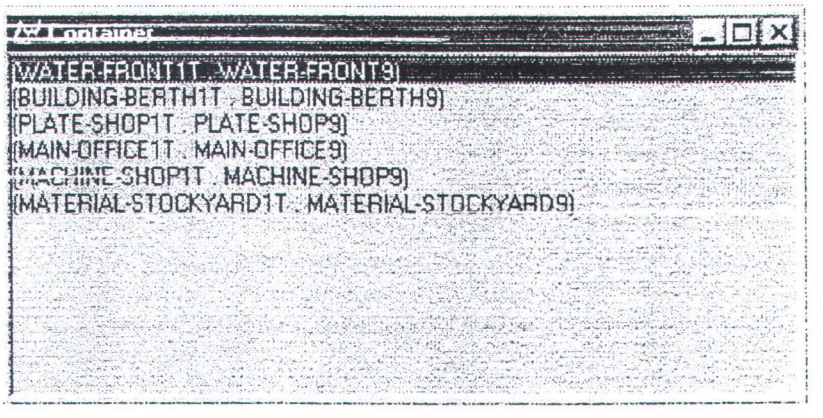
Layout Template No. 3
Date : 11 Juni 2001
Irawan Prasetyo

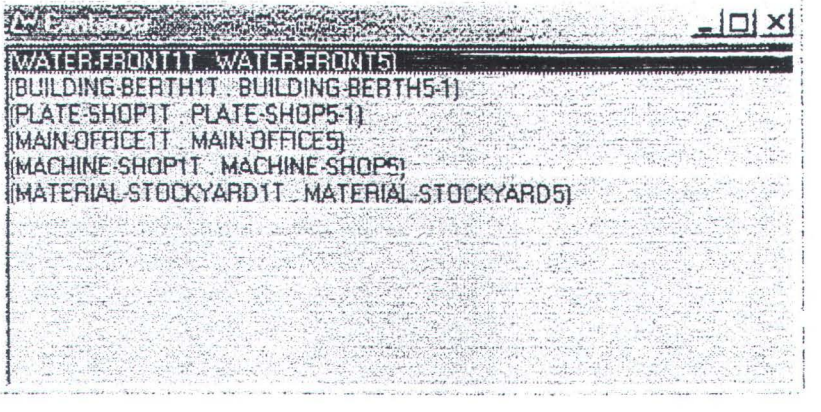
HASIL *MATCHING* ANTARA LAYOUT *TEMPLATE 1* DENGAN DATA *LAYOUT*


Container
Ordered layout similarity measures
(SHIPYARD-6 . SHIPYARD-TEMPLATE-1) (7 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-9 . SHIPYARD-TEMPLATE-1) (6 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-5 . SHIPYARD-TEMPLATE-1) (6 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-4 . SHIPYARD-TEMPLATE-1) (6 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-2 . SHIPYARD-TEMPLATE-1) (6 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-1 . SHIPYARD-TEMPLATE-1) (6 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-10 . SHIPYARD-TEMPLATE-1) (5 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-7 . SHIPYARD-TEMPLATE-1) (5 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-3 . SHIPYARD-TEMPLATE-1) (5 (WATER-FRONT1T . WATER-FRO
((SHIPYARD-8 . SHIPYARD-TEMPLATE-1) (4 (WATER-FRONT1T . WATER-FRO

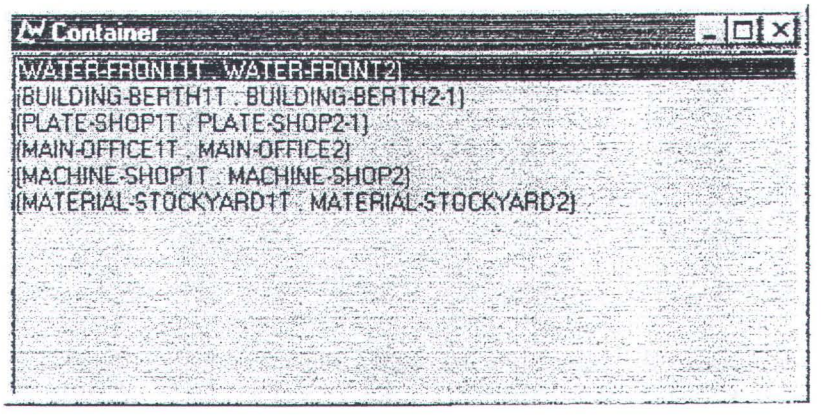
HASIL *MATCHING* ANTARA LAYOUT *TEMPLATE 1* DENGAN DATA *LAYOUT* SECARA LENGKAP


(WATER-FRONT1T . WATER-FRONT6)
(BUILDING-BERTH1T . BUILDING-BERTH6)
(PLATE-SHOP1T . PLATE-SHOP6)
(MAIN-OFFICE1T . MAIN-OFFICE6)
(MACHINE-SHOP1T . MACHINE-SHOP6-2)
(PIPING-SHOP1T . PIPING-SHOP6)
(MATERIAL-STOCKYARD1T . MATERIAL-STOCKYARD6-1)

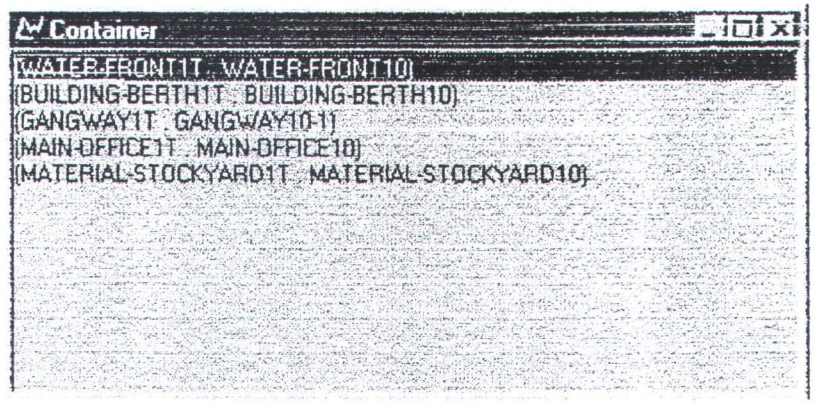

(WATER-FRONT1T . WATER-FRONT9)
(BUILDING-BERTH1T . BUILDING-BERTH9)
(PLATE-SHOP1T . PLATE-SHOP9)
(MAIN-OFFICE1T . MAIN-OFFICE9)
(MACHINE-SHOP1T . MACHINE-SHOP9)
(MATERIAL-STOCKYARD1T . MATERIAL-STOCKYARD9)

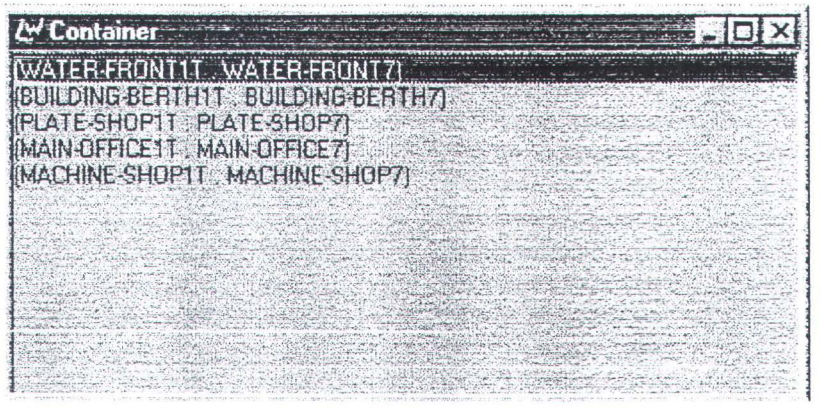

(WATER-FRONT1T . WATER-FRONT5)
(BUILDING-BERTH1T . BUILDING-BERTH5-1)
(PLATE-SHOP1T . PLATE-SHOP5-1)
(MAIN-OFFICE1T . MAIN-OFFICE5)
(MACHINE-SHOP1T . MACHINE-SHOP5)
(MATERIAL-STOCKYARD1T . MATERIAL-STOCKYARD5)

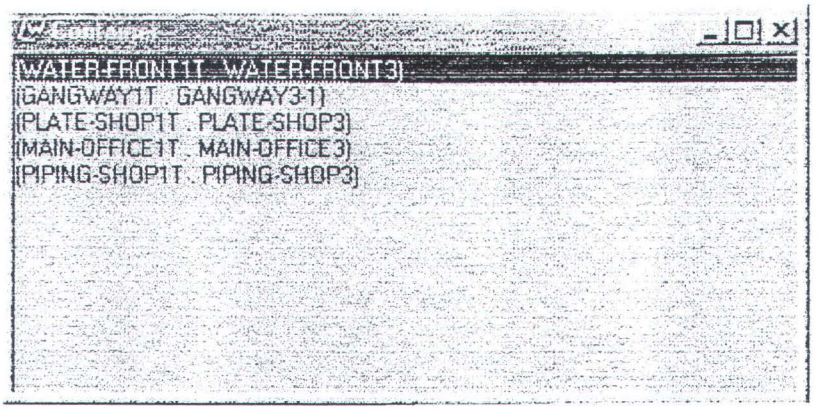
(WATER-FRONT1T . WATER-FRONT4)
(BUILDING-BERTH1T . BUILDING-BERTH4)
(PLATE-SHOP1T . PLATE-SHOP4)
(MAIN-OFFICE1T . MAIN-OFFICE4)
(MACHINE-SHOP1T . MACHINE-SHOP4)
(MATERIAL-STOCKYARD1T . MATERIAL-STOCKYARD4)


Container
(WATER-FRONT1T . WATER-FRONT2)
(BUILDING-BERTH1T . BUILDING-BERTH2-1)
(PLATE-SHOP1T . PLATE-SHOP2-1)
(MAIN-OFFICE1T . MAIN-OFFICE2)
(MACHINE-SHOP1T . MACHINE-SHOP2)
(MATERIAL-STOCKYARD1T . MATERIAL-STOCKYARD2)

(WATER-FRONT1T . WATER-FRONT1)
(BUILDING-BERTH1T . BUILDING-BERTH1)
(PLATE-SHOP1T . PLATE-SHOP1)
(MAIN-OFFICE1T . MAIN-OFFICE1)
(MACHINE-SHOP1T . MACHINE-SHOP1)
(PIPING-SHOP1T . PIPING-SHOP1)

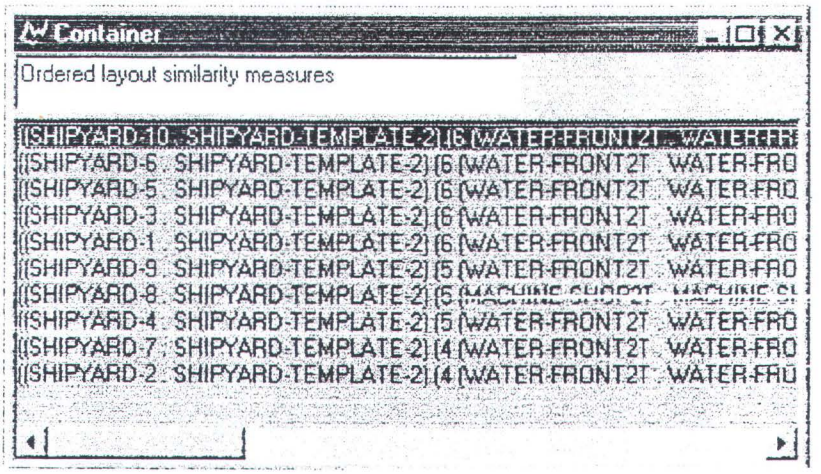

Container
(WATER-FRONT1T . WATER-FRONT10)
(BUILDING-BERTH1T . BUILDING-BERTH10)
(GANGWAY1T . GANGWAY10-1)
(MAIN-OFFICE1T . MAIN-OFFICE10)
(MATERIAL-STOCKYARD1T . MATERIAL-STOCKYARD10)


Container
(WATER-FRONT1T . WATER-FRONT7)
(BUILDING-BERTH1T . BUILDING-BERTH7)
(PLATE-SHOP1T . PLATE-SHOP7)
(MAIN-OFFICE1T . MAIN-OFFICE7)
(MACHINE-SHOP1T . MACHINE-SHOP7)

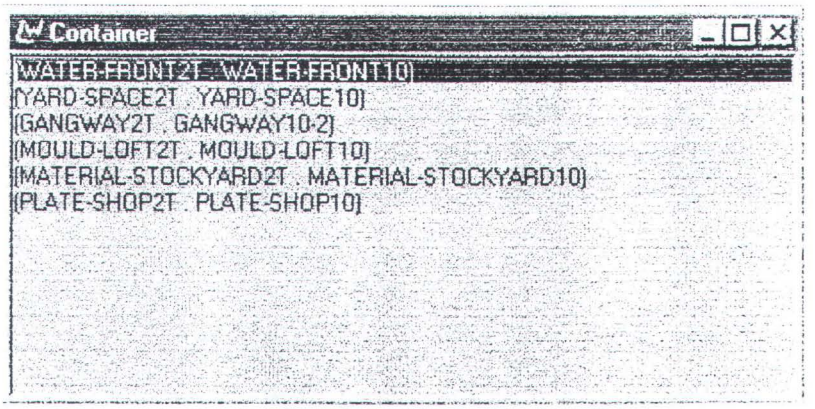

(WATER-FRONT1T . WATER-FRONT3)
(GANGWAY1T . GANGWAY3-1)
(PLATE-SHOP1T . PLATE-SHOP3)
(MAIN-OFFICE1T . MAIN-OFFICE3)
(PIPING-SHOP1T . PIPING-SHOP3)

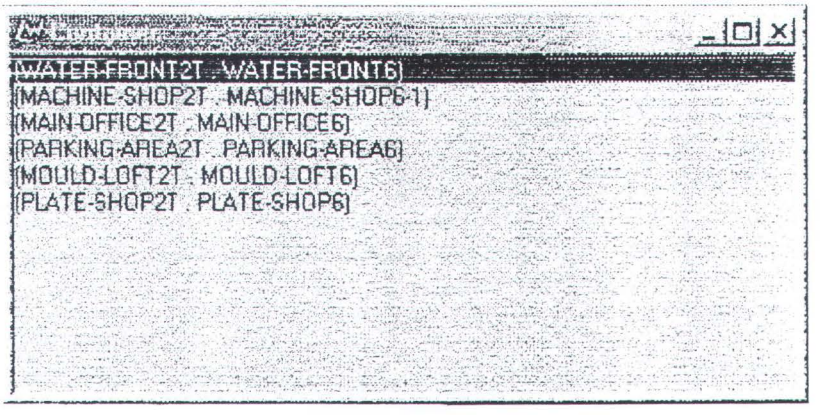
Container
(WATER-FRONT1T . WATER-FRONT8)
(GANGWAY1T . GANGWAY8-2)
(MOULD-LOFT1T . MOULD-LOFT8)
(MATERIAL-STOCKYARD1T . MATERIAL-STOCKYARD8)

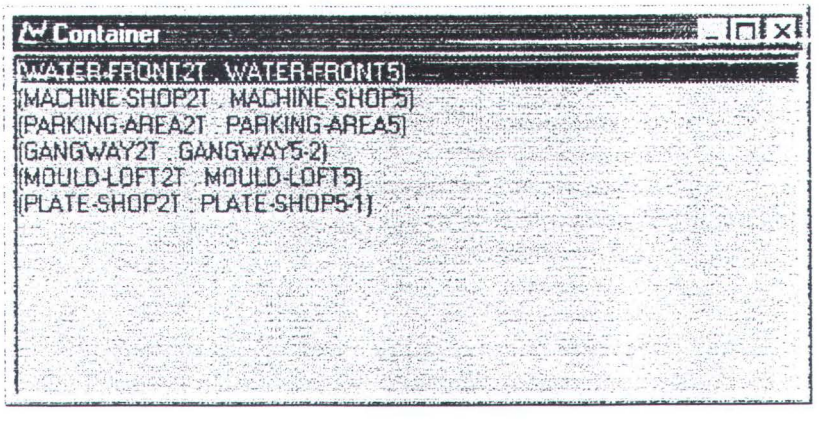
HASIL *MATCHING* ANTARA LAYOUT *TEMPLATE 2* DENGAN DATA *LAYOUT*

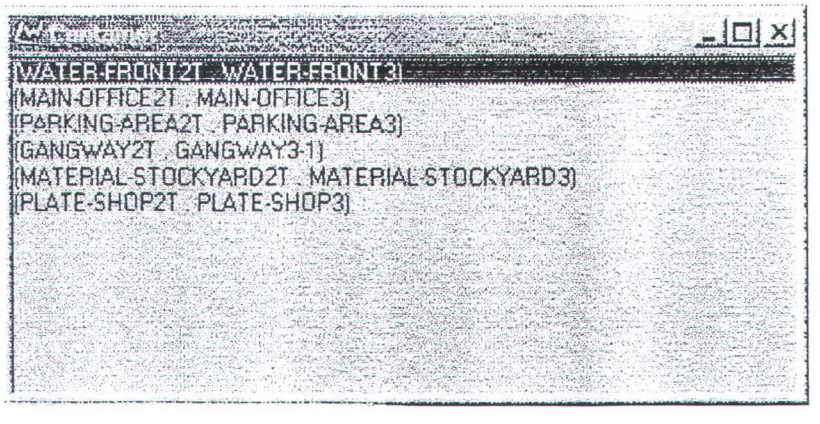

Container
Ordered layout similarity measures
((SHIPYARD-10 . SHIPYARD-TEMPLATE-2) (6 (WATER-FRONT2T . WATER-FR
((SHIPYARD-6 . SHIPYARD-TEMPLATE-2) (6 (WATER-FRONT2T . WATER-FRO
((SHIPYARD-5 . SHIPYARD-TEMPLATE-2) (6 (WATER-FRONT2T . WATER-FRO
((SHIPYARD-3 . SHIPYARD-TEMPLATE-2) (6 (WATER-FRONT2T . WATER-FRO
((SHIPYARD-1 . SHIPYARD-TEMPLATE-2) (6 (WATER-FRONT2T . WATER-FRO
((SHIPYARD-9 . SHIPYARD-TEMPLATE-2) (5 (WATER-FRONT2T . WATER-FRO
((SHIPYARD-8 . SHIPYARD-TEMPLATE-2) (5 (MACHINE-SHOP2T . MACHINE-SI
((SHIPYARD-4 . SHIPYARD-TEMPLATE-2) (5 (WATER-FRONT2T . WATER-FRO
((SHIPYARD-7 . SHIPYARD-TEMPLATE-2) (4 (WATER-FRONT2T . WATER-FRO
((SHIPYARD-2 . SHIPYARD-TEMPLATE-2) (4 (WATER-FRONT2T . WATER-FRO

HASIL *MATCHING* ANTARA LAYOUT *TEMPLATE 2* DENGAN DATA *LAYOUT* SECARA LENGKAP

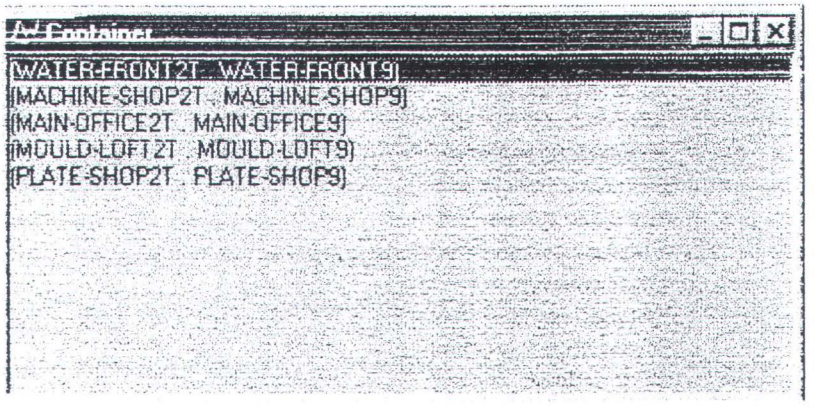

Container
(WATER-FRONT2T . WATER-FRONT10)
(YARD-SPACE2T . YARD-SPACE10)
(GANGWAY2T . GANGWAY10-2)
(MOULD-LOFT2T . MOULD-LOFT10)
(MATERIAL-STOCKYARD2T . MATERIAL-STOCKYARD10)
(PLATE-SHOP2T . PLATE-SHOP10)

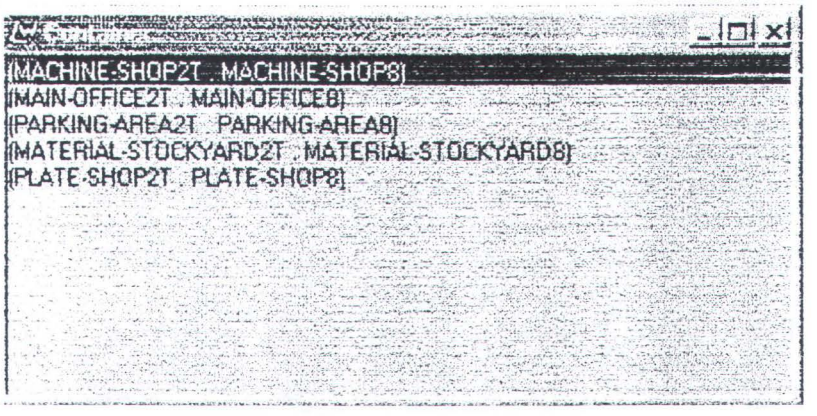

(WATER-FRONT2T . WATER-FRONT6)
(MACHINE-SHOP2T . MACHINE-SHOP6-1)
(MAIN-OFFICE2T . MAIN-OFFICE6)
(PARKING-AREA2T . PARKING-AREA6)
(MOULD-LOFT2T . MOULD-LOFT6)
(PLATE-SHOP2T . PLATE-SHOP6)

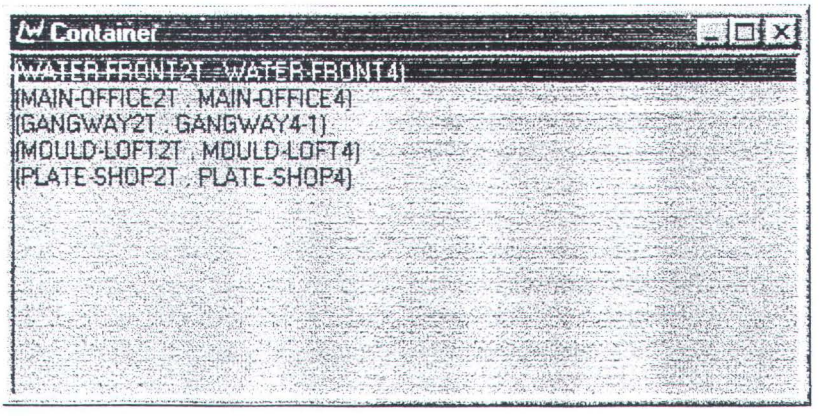

Container
(WATER-FRONT2T . WATER-FRONT5)
(MACHINE-SHOP2T . MACHINE-SHOP5)
(PARKING-AREA2T . PARKING-AREA5)
(GANGWAY2T . GANGWAY5-2)
(MOULD-LOFT2T . MOULD-LOFT5)
(PLATE-SHOP2T . PLATE-SHOP5-1)

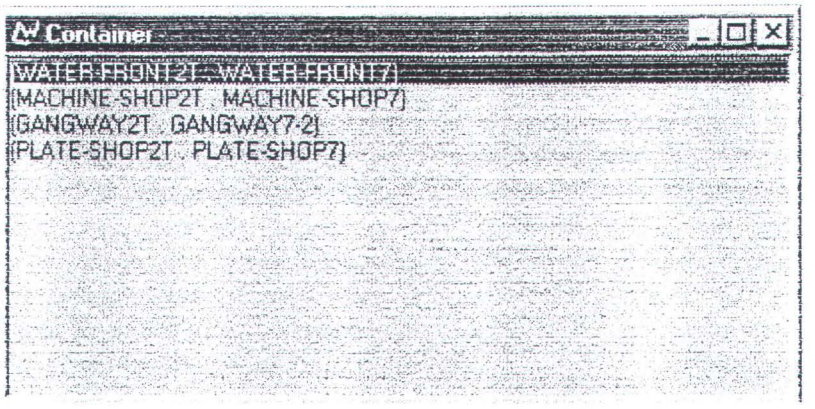

(WATER-FRONT2T . WATER-FRONT3)
(MAIN-OFFICE2T . MAIN-OFFICE3)
(PARKING-AREA2T . PARKING-AREA3)
(GANGWAY2T . GANGWAY3-1)
(MATERIAL-STOCKYARD2T . MATERIAL-STOCKYARD3)
(PLATE-SHOP2T . PLATE-SHOP3)

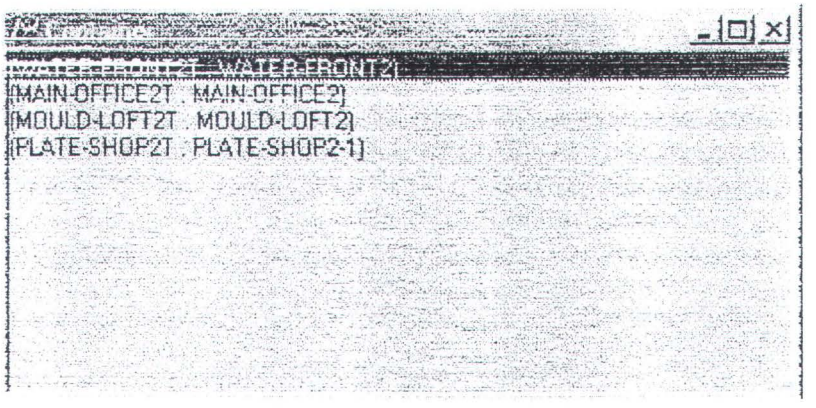
(WATER-FRONT2T . WATER-FRONT1)
(MAIN-OFFICE2T . MAIN-OFFICE1)
(PARKING-AREA2T . PARKING-AREA1)
(GANGWAY2T . GANGWAY1-1)
(MOULD-LOFT2T . MOULD-LOFT1)
(PLATE-SHOP2T . PLATE-SHOP1)


Container
(WATER-FRONT2T . WATER-FRONT9)
(MACHINE-SHOP2T . MACHINE-SHOP9)
(MAIN-OFFICE2T . MAIN-OFFICE9)
(MOULD-LOFT2T . MOULD-LOFT9)
(PLATE-SHOP2T . PLATE-SHOP9)

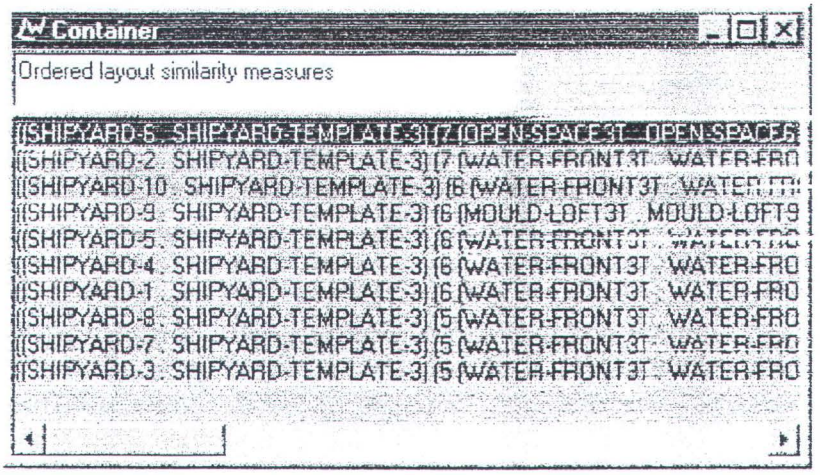

(MACHINE-SHOP2T . MACHINE-SHOP8)
(MAIN-OFFICE2T . MAIN-OFFICE8)
(PARKING-AREA2T . PARKING-AREA8)
(MATERIAL-STOCKYARD2T . MATERIAL-STOCKYARD8)
(PLATE-SHOP2T . PLATE-SHOP8)


Container
(WATER-FRONT2T . WATER-FRONT4)
(MAIN-OFFICE2T . MAIN-OFFICE4)
(GANGWAY2T . GANGWAY4-1)
(MOULD-LOFT2T . MOULD-LOFT4)
(PLATE-SHOP2T . PLATE-SHOP4)

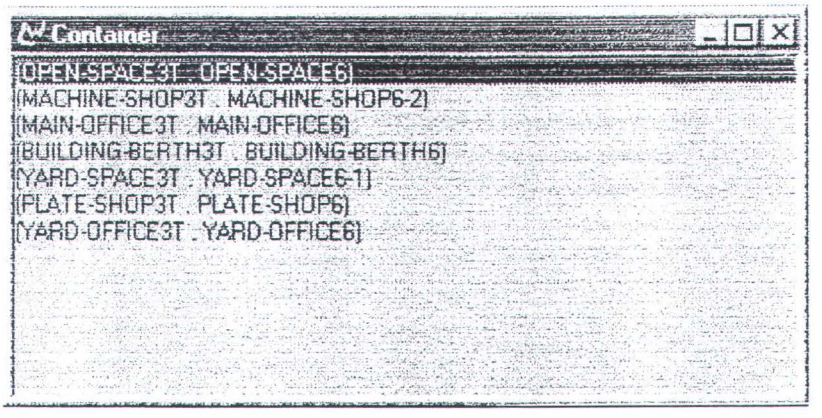

Container
(WATER-FRONT2T . WATER-FRONT7)
(MACHINE-SHOP2T . MACHINE-SHOP7)
(GANGWAY2T . GANGWAY7-2)
(PLATE-SHOP2T . PLATE-SHOP7)

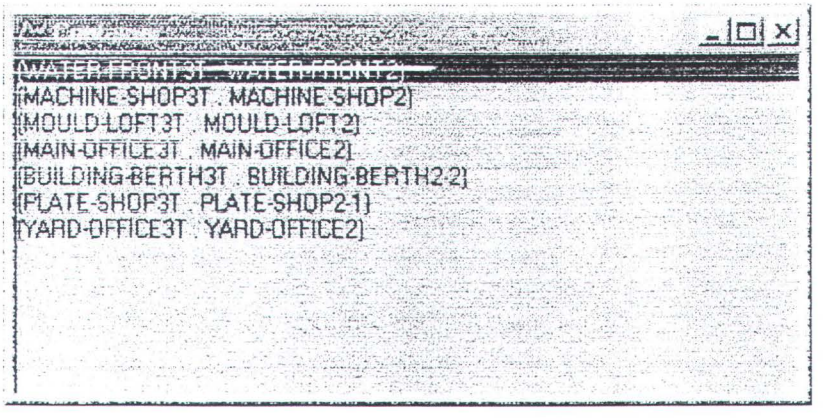

(MAIN-OFFICE2T . MAIN-OFFICE2)
(MOULD-LOFT2T . MOULD-LOFT2)
(PLATE-SHOP2T . PLATE-SHOP2-1)

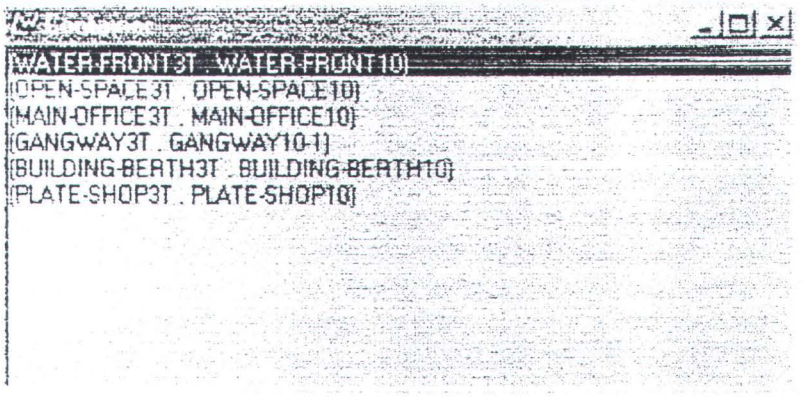
HASIL *MATCHING* ANTARA LAYOUT *TEMPLATE 3* DENGAN DATA *LAYOUT*

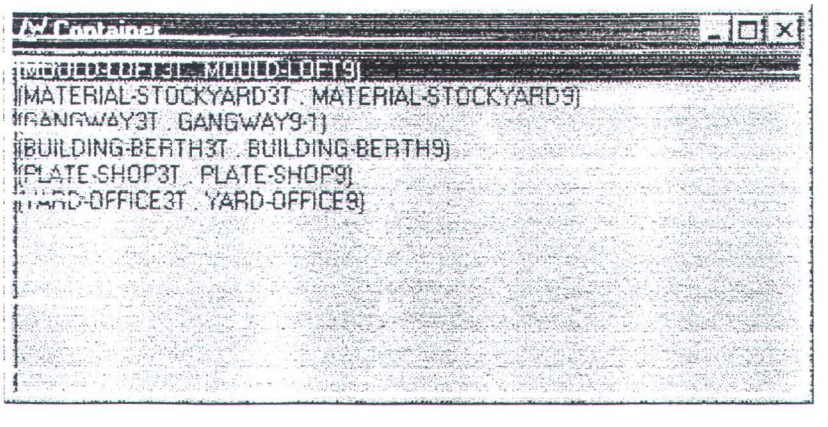

Container
Ordered layout similarity measures
((SHIPYARD-5 . SHIPYARD-TEMPLATE-3) (7 (OPEN-SPACE3T . OPEN-SPACE3
((SHIPYARD-2 . SHIPYARD-TEMPLATE-3) (7 (WATER-FRONT3T . WATER-FRO
((SHIPYARD-10 . SHIPYARD-TEMPLATE-3) (6 (WATER-FRONT3T . WATER-FR
((SHIPYARD-9 . SHIPYARD-TEMPLATE-3) (6 (MOULD-LOFT3T . MOULD-LOFT3
((SHIPYARD-5 . SHIPYARD-TEMPLATE-3) (6 (WATER-FRONT3T . WATER-FRO
((SHIPYARD-4 . SHIPYARD-TEMPLATE-3) (6 (WATER-FRONT3T . WATER-FRO
((SHIPYARD-1 . SHIPYARD-TEMPLATE-3) (6 (WATER-FRONT3T . WATER-FRO
((SHIPYARD-8 . SHIPYARD-TEMPLATE-3) (5 (WATER-FRONT3T . WATER-FRO
((SHIPYARD-7 . SHIPYARD-TEMPLATE-3) (5 (WATER-FRONT3T . WATER-FRO
((SHIPYARD-3 . SHIPYARD-TEMPLATE-3) (5 (WATER-FRONT3T . WATER-FRO

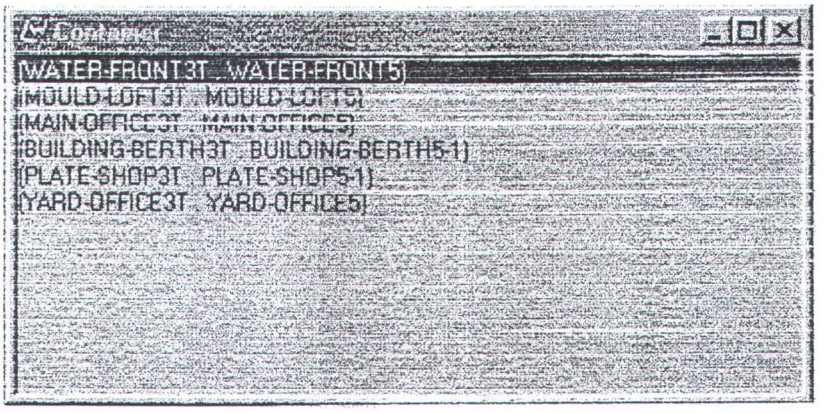
HASIL *MATCHING* ANTARA LAYOUT *TEMPLATE 3* DENGAN DATA *LAYOUT* SECARA LENGKAP

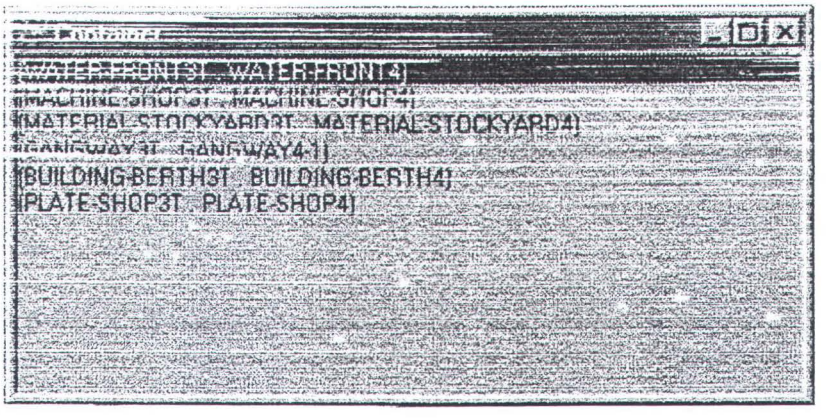

Container
(OPEN-SPACE3T . OPEN-SPACE6)
(MACHINE-SHOP3T . MACHINE-SHOP6-2)
(MAIN-OFFICE3T . MAIN-OFFICE6)
(BUILDING-BERTH3T . BUILDING-BERTH6)
(YARD-SPACE3T . YARD-SPACE6-1)
(PLATE-SHOP3T . PLATE-SHOP6)
(YARD-OFFICE3T . YARD-OFFICE6)

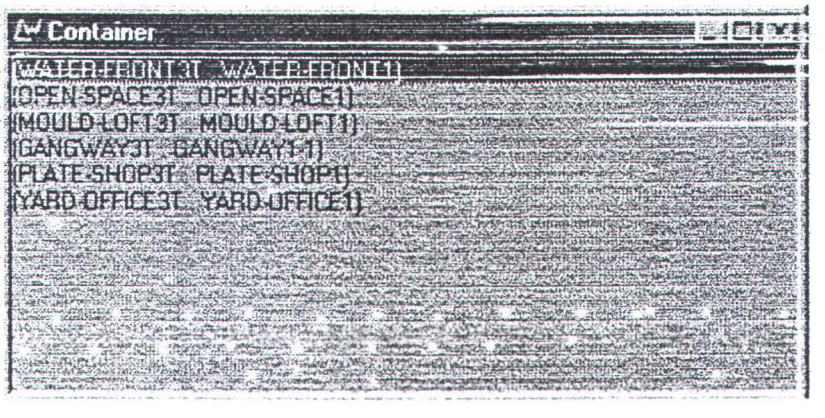

(MACHINE-SHOP3T . MACHINE-SHOP2)
(MOULD-LOFT3T . MOULD-LOFT2)
(MAIN-OFFICE3T . MAIN-OFFICE2)
(BUILDING-BERTH3T . BUILDING-BERTH2-2)
(PLATE-SHOP3T . PLATE-SHOP2-1)
(YARD-OFFICE3T . YARD-OFFICE2)


(WATER-FRONT3T . WATER-FRONT10)
(OPEN-SPACE3T . OPEN-SPACE10)
(MAIN-OFFICE3T . MAIN-OFFICE10)
(GANGWAY3T . GANGWAY10-1)
(BUILDING-BERTH3T . BUILDING-BERTH10)
(PLATE-SHOP3T . PLATE-SHOP10)

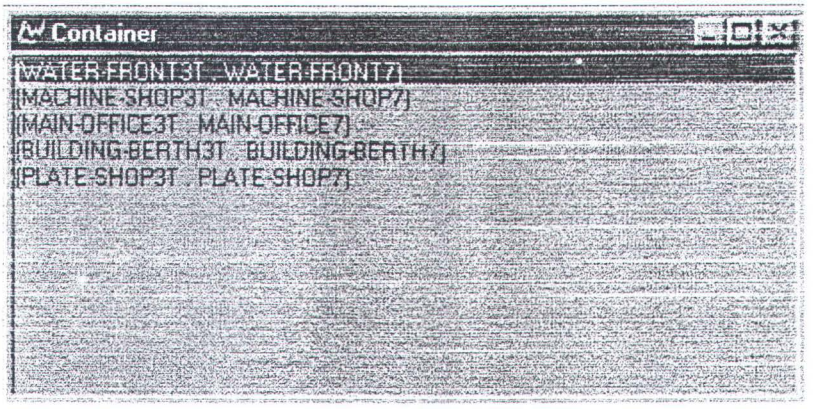

Container
(MATERIAL-STOCKYARD3T . MATERIAL-STOCKYARD9)
(GANGWAY3T . GANGWAY9-1)
(BUILDING-BERTH3T . BUILDING-BERTH9)
(PLATE-SHOP3T . PLATE-SHOP9)
(YARD-OFFICE3T . YARD-OFFICE8)

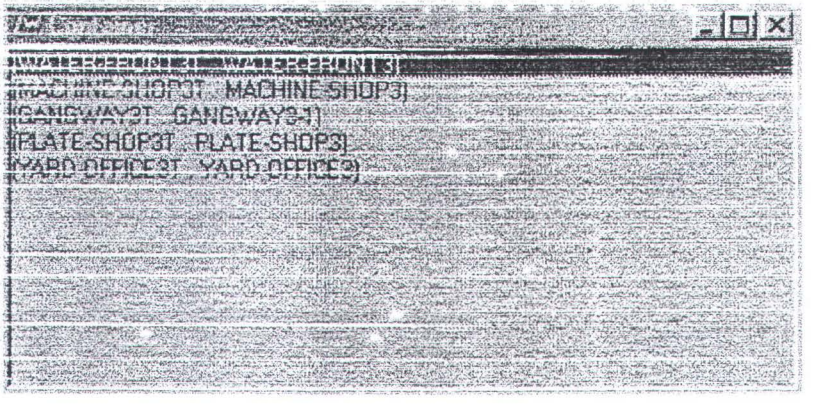

(WATER-FRONT3T . WATER-FRONT5)
(MOULD-LOFT3T . MOULD-LOFT5)
(BUILDING-BERTH3T . BUILDING-BERTH5-1)
(PLATE-SHOP3T . PLATE-SHOP5-1)
(YARD-OFFICE3T . YARD-OFFICE5)


(BUILDING-BERTH3T . BUILDING-BERTH4)
(PLATE-SHOP3T . PLATE-SHOP4)


Container
(WATER-FRONT3T . WATER-FRONT1)
(OPEN-SPACE3T . OPEN-SPACE1)
(MOULD-LOFT3T . MOULD-LOFT1)
(GANGWAY3T . GANGWAY1-1)
(PLATE-SHOP3T . PLATE-SHOP1)
(YARD-OFFICE3T . YARD-OFFICE1)

(WATER-FRONT3T . WATER-FRONT8)
(OPEN-SPACE3T . OPEN-SPACE8)
(MOULD-LOFT3T . MOULD-LOFT8)
(MAIN-OFFICE3T . MAIN-OFFICE8)
(PLATE-SHOP3T . PLATE-SHOP8)


Container
(WATER-FRONT3T . WATER-FRONT7)
(MACHINE-SHOP3T . MACHINE-SHOP7)
(MAIN-OFFICE3T . MAIN-OFFICE7)
(BUILDING-BERTH3T . BUILDING-BERTH7)
(PLATE-SHOP3T . PLATE-SHOP7)


(MACHINE-SHOP3T . MACHINE-SHOP3)
(GANGWAY3T . GANGWAY3-1)
(PLATE-SHOP3T . PLATE-SHOP3)

DEPARTEMEN PENDIDIKAN NASIONAL
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
FAKULTAS TEKNOLOGI KELAUTAN
JURUSAN TEKNIK PERKAPALAN
Kampus ITS -Sukolilo, Surabaya 60111 Telp. 5947254, 5994251-5 Pes. 1173 – 1176 Fax 5947254

SURAT KEPUTUSAN TUGAS AKHIR

No. : 410 b / K03.4.2/PP/2000

Nama Mahasiswa	: Irawan Prasetyo
Nomor Pokok	: 4198100511
Tanggal diberi tugas	: 01 September 2000
Tanggal selesai tugas	: 05 Januari 2001
Dosen Pembimbing	: 1. Ir. Djauhar Manfaat, MSc., Ph.D.
	2.

Uraian / judul tugas akhir yang diberikan :

#PENGGUNAAN TEKNIK TOPOLOGICAL PATTERN (GRAPH) MATCHING DALAM PEMILIHAN DESAIN TATA LETAK GALANGAN KAPAL#

Surabaya, 20 September 2000

Jurusan Teknik Perkapalan
K e t u a,

Ir. Djauhar Manfaat, MSc., Ph.D.
NIP. 131 651 444.

Tembusan :
1. Yth. Dekan FTK-ITS
2. Yth. Dosen Pembimbing
3. Arsip

FAKULTAS TEKNOLOGI KELAUTAN ITS
JURUSAN TEKNIK PERKAPALAN

DAFTAR KEMAJUAN TUGAS AKHIR (NA 1701)

mahasiswa : Irawan Prasetyo
: 4198100511
diberikan : Semester ~~Genap~~ Gasal 2000/2001
al mulai tugas : 01 September 2000
al selesai tugas : 05 Januari 2001
Pembimbing : 1. Ir. Djauhar Manfaat, MSc, Ph.D.
2. .....

ggal	Uraian Kemajuan Tugas	Tanda Tangan
2000	Diskusi awal ttg. T.A.	
-2000	: Penjelasan ttg. Metode Pattern Matching yg. akan digunakan. - Bab I draft sudah diserahkan.	
.2000	- Bab II draft sudah diserahkan - Diskusi tentang Bab I & II	
2000	- Diskusi tentang rencana implementasi program komputer	
2001	- Bab III draft untuk diserahkan - Bab II & III supaya direvisi - Demo program komputer	
2001	- Demo program simpsi dengan space clearance	
001	- Interface dari data layout yang digunakan	
001	- Revisi bab III	

lihat halaman berikutnya .....

Tanggal	Uraian Kemajuan Tugas	Tanda Tangan
25/4'2001	Melanjutkan interface program	
3/5'2001	Interface dari layout template	
10/5'2001	Bab IV draft untuk diseminarkan	
26/5'2001	Melanjutkan interface dan sistem	
7/6'2001	Bab V draft untuk diseminarkan	
21/6'2001	Revisi bab IV & V	
28/6'2001	Pembuatan program. & pakem instalasi	
5/7'2001	Melanjutkan pembuatan program.	
9/7'2001	Draft TA untuk diseminarkan	

Catatan :
- Formulir ini harus dibawa pada saat konsultasi
- Konsultasi dilaksanakan minimal seminggu sekali.
- Formulir ini harus dikumpulkan kembali pada saat mengumpulkan laporan tugas akhir.

Surabaya, 17 Juli 2001
Dosen Pembimbing,

Dr. Djunaidi Manfaat
NIP. 131 651 444