

PEMODELAN DAN MANAJEMEN MODEL

Pemodelan dalam MSS.

Salah satu contoh DSS, yaitu dari Frazee Paint, Inc., memiliki 3 jenis model:

- 1. **Model statistik (analisis regresi)**, digunakan untuk mencari relasi diantara variabel. Model ini merupakan preprogram dalam tool software pengembangan DSS.
- 2. **Model finansial** untuk pengembangan laporan pemasukan dan proyeksi data finansial untuk beberapa tahun. Model ini semi terstruktur dan ditulis dalam bahasa khusus DSS yang disebut dengan IFPS.
- 3. **Model optimasi** yang dibuat menggunakan model management science yang disebut pendekatan Linear Programming dalam rangka menentukan pemilihan media. Untuk menggunakan model ini, DSS perlu antarmuka untuk berhubungan dengan software yang lain.

Berbagai aspek dalam pemodelan diantaranya adalah:

- Identifikasi masalah dan analisis lingkungan.
- Identifikasi variabel.
- Perkiraan (forecasting).
- Model.
- Manajemen model.

Di bawah ini adalah tabel Berbagai jenis model:

Category	Process and Objective	Representative Techniques
Optimization of problems with few alternatives	Find the best solution from a relatively small number of alternatives	Decision tables, decision trees
Optimization via algorithm	Find the best solution from a large or an infinite number of alternatives using a step-by-step improvement process	Linear and other mathematical programming models, network models
Optimization via analytical formula	Find the best solution, in one step, using a formula	Some inventory models
Simulation	Finding "good enough" solution, or the best among those alternatives checked, using experimentation	Several types of simulation
Heuristics	Find "good enough" solution using rules	Heuristic programming, expert systems
Other descriptive models	Finding "what-if" using a formula	Financial modelling, waiting lines
Predictive models	Predict future for a given scenario	Markov analysis, forecasting models

Model Statis dan Dinamis.

- **Analisis statis.** Model statis mengambil satu kejadian saja dalam suatu situasi. Selama kejadian tersebut semuanya terjadi dalam 1 interval, baik waktunya sebentar atau lama. Diasumsikan adanya stabilitas disini.
- **Analisis dinamis.** Model dinamis digunakan untuk mengevaluasi skenario yang berubah tiap saat. Model ini tergantung pada waktu. Dapat menunjukkan tren dan pola pada waktu tertentu.

Pohon Keputusan.

Alternatif penampilan tabel keputusan adalah pohon keputusan. Pohon keputusan memiliki 2 keuntungan: pertama, menggambarkan secara grafis hubungan dari masalah, dan kedua, dapat berhubungan dengan situasi yang lebih kompleks dalam bentuk yang lebih kompak (misal masalah investasi dengan periode waktu yang lebih banyak).

Metode mengatasi resiko yang lain.

Misalnya: simulasi, certainty factors, dan fuzzy logic.

Optimasi dengan Pemrograman Matematis.

Pemrograman matematis.

Digunakan untuk membantu menyelesaikan masalah manajerial, untuk mengalokasikan resources yang terbatas (misal tenaga kerja, modal, mesin, atau air) diantara sekian banyak aktivitas untuk mengoptimalkan tujuan yang ditetapkan.

Karakteristik.

1. Sejumlah tertentu resources ekonomi tersedia untuk dialokasi.
2. Resources digunakan dalam produksi produk atau service.
3. Ada 2 atau lebih cara bagaimana resources digunakan. Masing-masingnya disebut dengan solusi atau program.
4. Setiap aktivitas (produk atau service) dimana resources digunakan disitu memberikan hasil tertentu sesuai tujuan yang telah ditetapkan.
5. Pengalokasian ini biasanya dibatasi oleh Berbagai batasan dan kebutuhan yang disebut dengan constraints (batasan).

Asumsi.

1. Hasil dari Berbagai alokasi yang berbeda dapat dibandingkan; sehingga, mereka dapat diukur dengan unit yang sama (seperti dolar atau utilitas).
2. Hasil dari Berbagai alokasi berdiri sendiri dibandingkan dengan alokasi yang lain.
3. Hasil total adalah penjumlahan dari semua hasil yang diperoleh dari aktivitas-aktivitas yang berbeda.
4. Semua data diketahui dengan certainty.
5. Resources digunakan menurut perilaku ekonomi.

Penggunaan pemrograman matematis ini, khususnya Linear Programming, begitu umumnya sehingga melingkupi program-program komputer yang ada pada setiap organisasi.

Linear Programming (LP).

Blending Problem (minimisasi).

Disajikan contoh dari LP tersebut, yang dikenal dengan blending problem (masalah pengenceran). Untuk membuat cat Sungold, dibutuhkan cat yang memiliki tingkat brilliance paling tidak 300 derajat dan level hue paling tidak 250 derajat. Level brilliance dan hue ditentukan oleh 2 formula, Alpha dan Beta. Baik Alpha dan Beta memberikan kontribusi yang sama ke tingkat brilliance yang dibutuhkan; 1 ounce (berat kering) dari keduanya menghasilkan 1 derajat brilliance dalam 1 drum cat. Namun demikian, hue diatur seluruhnya oleh jumlah Alpha-nya; 1 ounce darinya menghasilkan 3 derajat hue dalam 1 drum cat. Biaya Alpha adalah 45 cents per ounce, dan biaya Beta adalah 12 cent per ounce. Diasumsikan bahwa tujuan dari kasus ini adalah meminimalkan biaya resources, maka masalahnya adalah untuk menemukan jumlah Alpha dan Beta yang harus dipenuhi untuk membuat setiap drum cat.

Perumusan Blending Problem.

Decision variables-nya adalah:

x_1 = jumlah Alpha yang diperlukan, dalam ounces, dalam setiap drum cat

x_2 = jumlah Beta yang diperlukan, dalam ounces, dalam setiap drum cat

Tujuannya adalah untuk meminimalkan biaya total dari formula yang dibutuhkan untuk menghasilkan 1 drum cat. Jika biaya Alpha adalah 45 cent per ounce, dan jika x_1 ounce digunakan dalam setiap drum, maka biaya per drum adalah $45x_1$. Serupa dengan itu, untuk Beta biayanya adalah $12x_2$. Biaya totalnya menjadi, $45x_1 + 12x_2$, dan fungsi tujuan kita, adalah untuk meminimisasikan hal-hal di atas berdasarkan batasan di bawah ini:

1. Untuk membuat tingkat brilliance paling tidak 300 derajat dalam setiap drum. Karena setiap ounce Alpha atau Beta meningkatkan derajat kecerahan (brightness) 1 derajat, maka terjadilah hubungan berikut:

$$\begin{array}{ccc} \text{Disuplai oleh Alpha} & \text{Disuplai oleh Beta} & \text{Permintaan} \\ \underbrace{\text{Di sup lai oleh Alpha}}_{1x_1} + \underbrace{\text{Di sup lai oleh Beta}}_{1x_2} & \geq & \underbrace{\text{Per min taan}}_{300} \end{array}$$

2. Untuk membuat level hue paling tidak 250 derajat, efek dari Alpha (sendirian) pada hue dapat ditulis sebagai berikut:

$$\underbrace{\text{Di sup lai oleh Alpha}}_{3x_1} + \underbrace{\text{Di sup lai oleh Beta}}_{0x_2} \geq \underbrace{\text{Per min taan}}_{250}$$

Ringkasnya blending problem diformulasikan seperti ini:

Temukan x_1 dan x_2 yang:

Mimenimisasikan $z = 45x_1 + 12x_2$

Dengan batasan:

$$1x_1 + 1x_2 \geq 300 \quad (\text{spesifikasi kecerahan, brightness})$$

$$3x_1 + 0x_2 \geq 250 \quad (\text{spesifikasi hue})$$

Solusi. (Dihasilkan oleh komputer)

$$X_1 = 83.333$$

$$X_2 = 216.667$$

$$\text{Biaya total} = \$63.50$$

Perumusan Umum dan Istilah.

Dibahas disini perumusan umum dari LP, dimana setiap LP terdiri dari:

Decision Variables.

Variabel-variabel dimana nilainya tak diketahui dan yang sedang dicari. Biasanya ditandai dengan x_1 , x_2 , dan lain-lain.

Objective Function (Fungsi Tujuan).

Pernyataan matematis, merupakan fungsi linier, menunjukkan hubungan diantara decision variables dan satu tujuan (atau objective) yang dicari. Jika melibatkan tujuan yang banyak (multiple goals), terdapat 2 pendekatan:

1. Memilih tujuan utama yang memiliki level maksimal atau minimal.
2. Memindahkan tujuan-tujuan yang lain ke dalam constraint (batasan), yang harus dipenuhi.

Optimasi.

LP berusaha untuk mendapatkan nilai maksimal atau minimal dari fungsi tujuan.

Coefficients (Koefisien) dari Objective Function.

Menyatakan tingkat/derajat dimana nilai dari fungsi tujuan meningkat atau menurun dengan memasukkan dalam solusi satu unit dari setiap decision variables.

Constraints (batasan).

Maksimalisasi atau minimalisasi dilakukan berdasarkan batasan-batasan tertentu. Sehingga, LP dapat didefinisikan sebagai permasalahan optimasi terbatas. Batasan dinyatakan dalam bentuk pertidaksamaan (atau terkadang persamaan).

Koefisien Input-Output (Teknologi)

Koefisien dari variabel batasan disebut dengan koefisien input-output. Ini mengindikasikan tingkat pemakaian atau penggunaan resource. Ditampilkan pada sisi kiri batasan.

Capacities (kapasitas).

Kapasitas (atau ketersediaan) dari Berbagai resources, biasanya dinyatakan dengan batas atas atau batas bawah, berada pada sisi kanan batasan. Sisi kanan juga menyatakan kebutuhan minimum.

Contoh.

Contoh dari perumusan umum dan istilah ini diterapkan pada blending problem sebelumnya. Temukan x_1 dan x_2 (decision variables) yang akan meminimisasikan nilai dari fungsi tujuan linier:

$$\begin{array}{c} \text{cost coefficients} \\ \downarrow \\ z = 45x_1 + 12x_2 \\ \uparrow \\ \text{decision variables} \end{array}$$

berdasarkan batasan-batasan linier:

$$\begin{array}{c} 1x_1 + 1x_2 \geq 300 \\ 3x_1 + 0x_2 \geq 250 \end{array}$$

\uparrow input-output coefficients \leftarrow capacities or requirements

Simulasi.

Dalam MSS artinya adalah teknik untuk melakukan percobaan (seperti misalnya “what-if”) dengan komputer digital pada suatu model dari sistem manajemen.

Karakteristik Utama.

Pertama, simulasi bukanlah sejenis model biasa; model umumnya merepresentasikan kenyataan, sedangkan simulasi biasanya menirukan kenyataan tersebut. Singkatnya, ini berarti ada sedikit penyederhanaan kenyataan dalam model simulasi dibandingkan dengan jenis model lainnya.

Kedua, simulasi adalah teknik untuk melaksanakan percobaan. Artinya, simulasi melibatkan testing pada nilai-nilai tertentu dari decision atau uncontrollable variables yang ada pada model dan mengamati akibatnya pada variabel output.

Simulasi lebih bersifat deskriptif (menjelaskan) daripada tool normatif; sehingga tak ada pencarian otomatis untuk solusi optimal. Lebih dari itu, simulasi menjelaskan dan/atau memperkirakan karakteristik sistem tertentu pada Berbagai keadaan yang berbeda-beda. Sekali karakteristik ini diketahui, alternatif terbaik dari alternatif yang ada dapat dipilih.

Simulasi digunakan bilamana permasalahan yang ada terlalu kompleks/sulit bila diselesaikan dengan teknik optimasi numerik (misalnya LP). Kompleksitas disini berarti bahwa permasalahan tadi tak bisa dirumuskan untuk optimasinya atau perumusannya terlalu kompleks.

Keuntungan Simulasi:

1. Teori simulasi relatif mudah dan bisa langsung diterapkan.
2. Model simulasi mudah untuk menggabungkan Berbagai hubungan dasar dan ketergantungannya.
3. Simulasi lebih bersifat deskriptif daripada normatif. Ini mengijinkan manajer untuk menanyakan jenis pertanyaan “what-if”. Sehingga, manajer yang memiliki pendekatan trial-and-error dalam menyelesaikan masalah dapat melakukannya lebih cepat dan murah, dengan resiko yang lebih kecil, menggunakan bantuan simulasi dan komputer (sebagai pembanding adalah pendekatan trial-and-error dalam sistem nyata).
4. Model simulasi yang akurat membutuhkan knowledge yang dalam dari suatu masalah, yang memaksa MSS builder untuk selalu berkomunikasi dengan manajer.
5. Modelnya dibangun berdasarkan perspektif manajer dan berada dalam struktur keputusannya.
6. Model simulasi dibangun untuk satu permasalahan tertentu, dan biasanya tak bisa menyelesaikan permasalahan yang lain.
7. Simulasi dapat mengatasi variasi yang berbeda-beda dalam Berbagai jenis masalah seperti halnya inventory dan staffing, demikian juga pada fungsi tingkat tinggi manajerial seperti rencana jangka panjang. Sehingga ungkapan untuknya adalah “selalu ada” jika manajer sedang membutuhkannya.

- 8. Manajer dapat melakukan eksperimen dengan Berbagai variabel yang berbeda untuk menentukan mana yang penting, dan dengan Berbagai alternatif yang berbeda untuk mencari yang terbaik.
- 9. Simulasi secara umum mengijinkan kita memasukkan kompleksitas kehidupan nyata dari suatu masalah; penyederhanaan tak diperlukan disini. Sebagai contoh, simulasi dapat memanfaatkan distribusi probabilitas kehidupan nyata daripada mengira-ira distribusi teoritis.
- 10. Sebagai sifat alamiah simulasi, kita dapat menghemat waktu.
- 11. Mudah untuk mendapatkan Berbagai pengukuran kinerja yang berbeda-beda secara langsung dari simulasi.

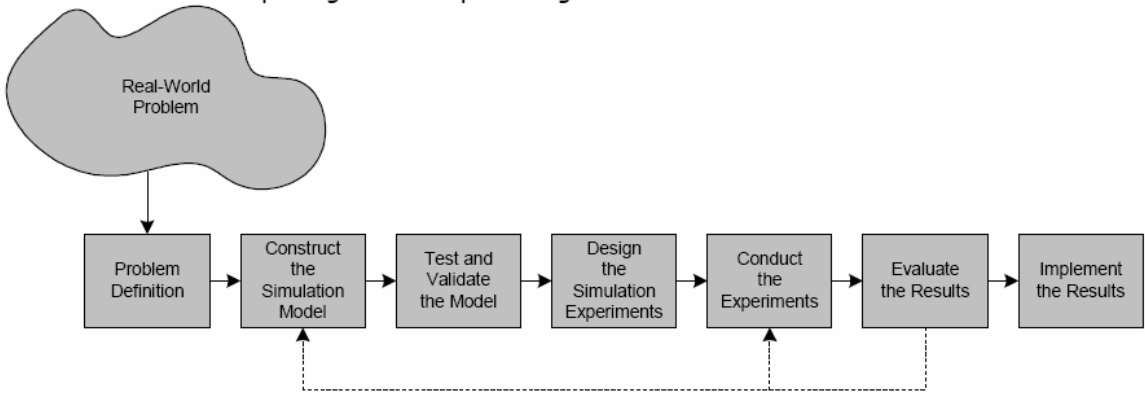
Kerugian Simulasi:

- 1. Tak menjamin solusi yang optimal.
- 2. Membangun model simulasi seringkali memakan waktu lama dan membutuhkan biaya.
- 3. Solusi dan inferensi dari satu kasus simulasi biasanya tak bisa ditransfer ke permasalahan yang lain.
- 4. Simulasi terkadang begitu mudah diterima oleh manajer sehingga solusi analitis yang dapat menghasilkan solusi optimal malah sering dilupakan.

Metodologi Simulasi.

- Definisi masalah.
- Membangun model simulasi.
- Testing dan validasi model.
- Desain percobaan.
- Melakukan percobaan.
- Evaluasi hasil.
- Implementasi.

Proses dari simulasi dapat digambarkan pada diagram di bawah ini:



Tipe Simulasi.

- **Simulasi Probabilistik.** Satu atau lebih independent variable-nya (misal, kebutuhan dalam masalah inventory) probabilistik, mengandung bilangan acak. Sehingga ini mengikutsertakan distribusi probabilitas tertentu. 2 yang dikenal: distribusi diskrit dan distribusi kontinyu. Distribusi diskrit melibatkan situasi dimana terdapat sejumlah tertentu kejadian (atau variabel) yang dapat diamati pada sejumlah nilai tertentu. Distribusi kontinyu mengacu pada situasi dimana terdapat kemungkinan jumlah kejadian yang tak terbatas, yang mengikuti fungsi densitas tertentu, misal distribusi normal. Di bawah ini dijelaskan bedanya:

Discrete		Continuous
Daily Demand	Probability	
5	0.10	Daily demand is normally distributed with a mean of 7 and a standard deviation 1.2
6	0.15	
7	0.30	
8	0.25	
9	0.20	

- Simulasi bergantung waktu (**time dependent**) vs simulasi tak bergantung waktu (**time independent**). Time independent mengacu pada situasi dimana tak penting kita mengetahui secara pasti kejadian yang terjadi. Time dependent sebaliknya, adalah penting mengetahui secara presisi kejadian-kejadiannya.
- **Simulasi visual.** Penampilan hasil simulasi secara grafis terkomputerisasi adalah salah satu pengembangan yang berhasil dalam interaksi manusia dan komputer dan penyelesaian masalah.

Percobaan dalam Simulasi (Probabilistik).

Proses percobaan dalam simulasi melibatkan 8 langkah, termasuk di dalamnya apa yang disebut dengan prosedur Monte Carlo:

1. Menentukan ukuran (ukuran-ukuran) kinerja sistem yang diinginkan. Jika perlu, tulislah dalam bentuk persamaan.
2. Gambarkan sistem dan dapatkan distribusi probabilitas dari elemen probabilistik yang relevan dari sistem.
3. Rancang bangunlah distribusi probabilistik kumulatif untuk setiap elemen stokastik.
4. Tentukan representasi bilangan yang berhubungan dengan distribusi probabilistik kumulatif.
5. Untuk setiap elemen probabilistik, ambil satu contoh acak (keluarkan bilangan secara acak atau ambil satu dari tabel bilangan acak).
6. Catat ukuran kinerja dan varian-nya.
7. Jika diinginkan hasil yang stabil, ulangi langkah 5 dan 6 sehingga ukuran kinerja sistem "stabil".
8. Ulangi langkah 5-7 untuk Berbagai alternatif. Berikan nilai ukuran kinerjanya dan confidence interval-nya, berdasarkan pada alternatif yang diinginkan.

Prosedure Monte Carlo bukanlah model simulasi, walaupun ia hampir menjadi sinonim dengan simulasi probabilistik. Prosedur tersebut melibatkan langkah 3 sampai langkah 6 dalam proses ini. Yaitu, prosedur yang menghasilkan pengamatan acak dari variabel-variabel yang penting.

Pemrograman Heuristic.

Pendekatan yang melibatkan cara heuristic (role of thumb, aturan jempol) yang dapat menghasilkan solusi yang layak dan cukup baik pada Berbagai permasalahan yang kompleks. Cukup baik (good enough) biasanya dalam jangkauan 90 sampai dengan 99.99 persen dari solusi optimal sebenarnya.

Metodologi.

Pendekatan lojik heuristic melibatkan hal-hal berikut:

1. Skema klasifikasi yang mengenalkan struktur ke dalam permasalahan.
2. Analisis karakteristik dari elemen-elemen masalah.
3. Aturan-aturan untuk seleksi elemen dari setiap kategori untuk mendapatkan strategi pencarian yang efisien.
4. Aturan-aturan untuk seleksi lebih lanjut, bila diperlukan.
5. Fungsi tujuan yang digunakan untuk mengecek kelayakan solusi pada setiap tahapan seleksi atau pencarian.

Kapan Menggunakan Heuristic:

1. Input data tidak pasti atau terbatas.
2. Kenyataan yang ada terlalu kompleks sehingga model optimasi menjadi terlalu disederhanakan.
3. Metode yang handal dan pasti tak tersedia.
4. Waktu komputasi untuk optimasi terlalu lama.
5. Adanya kemungkinan untuk meningkatkan efisiensi proses optimasi (misal, dengan memberikan solusi awal yang baik menggunakan heuristic).
6. Masalah-masalah yang diselesaikan seringkali (dan berulang-ulang) dan menghabiskan waktu komputasi.

7. Permasalahan yang kompleks yang tidak ekonomis untuk optimasi atau memakan waktu terlalu lama dan heuristic dapat meningkatkan solusi yang tak terkomputerisasi.
8. Di saat pemrosesan simbolik lebih banyak dilibatkan daripada pemrosesan numerik (dalam ES).

Keuntungan Heuristic:

1. Mudah dimengerti dan kemudian mudah diimplementasikan.
2. Membantu dalam melatih orang sehingga kreatif dan dapat digunakan untuk masalah yang lain.
3. Menghemat waktu perumusan.
4. Menghemat pemrograman dan kebutuhan penyimpanan pada komputer.
5. Menghemat waktu pemrosesan komputer yang tak perlu (kecepatan!).
6. Seringkali menghasilkan Berbagai solusi yang dapat diterima.

Masalah-masalah dalam Penggunaan Heuristic:

1. Heuristic enumerasi yang mempertimbangkan semua kemungkinan kombinasi dalam permasalahan praktis jarang bisa dicapai.
2. Pilihan-pilihan keputusan sekuensial bisa jadi gagal mengantisipasi konsekuensi lebih lanjut dari setiap pilihan.
3. "Lokal optimal" dapat memutuskan solusi terbaik yang masih bisa dicapai sebab heuristic, serupa dengan simulasi, bertitik tolak pada perspektif global.
4. Saling ketergantungan pada satu bagian dari sistem terkadang memberikan pengaruh berarti

Kesimpulan.

- Model memainkan peranan yang utama dalam DSS. Terdiri dari beberapa jenis.
- Manajemen model adalah konsep yang serupa dengan manajemen data.
- Model bisa berupa statis (cuplikan singkat situasi) atau dinamis.
- Analisis dilakukan baik pada kondisi certainty/kepastian (yang paling diinginkan memang ini), resiko, atau uncertainty (dihindari sebisanya).
- Tool utama dalam optimasi adalah pemrograman matematis.
- LP adalah tool paling sederhana dari pemrograman matematis. Ia mencoba mencari alokasi optimal dari resources yang terbatas pada batasan-batasan (constraints) yang ada.
- Bagian utama dari LP adalah objective function, decision variables, dan constraints.
- Simulasi banyak digunakan dalam pendekatan DSS yang melibatkan eksperimen dengan model yang diasumsikan merupakan pencerminan kenyataan yang sebenarnya.
- Simulasi dapat dilakukan untuk situasi yang kompleks, yang tak bisa dilakukan oleh teknik optimasi biasa. Tapi tak ada jaminan untuk mendapatkan solusi optimal.
- Pemrograman heuristic adalah penyelesaian masalah menggunakan rule of thumb.

PENGEMBANGAN SISTEM PENDUKUNG KEPUTUSAN

Pendahuluan.

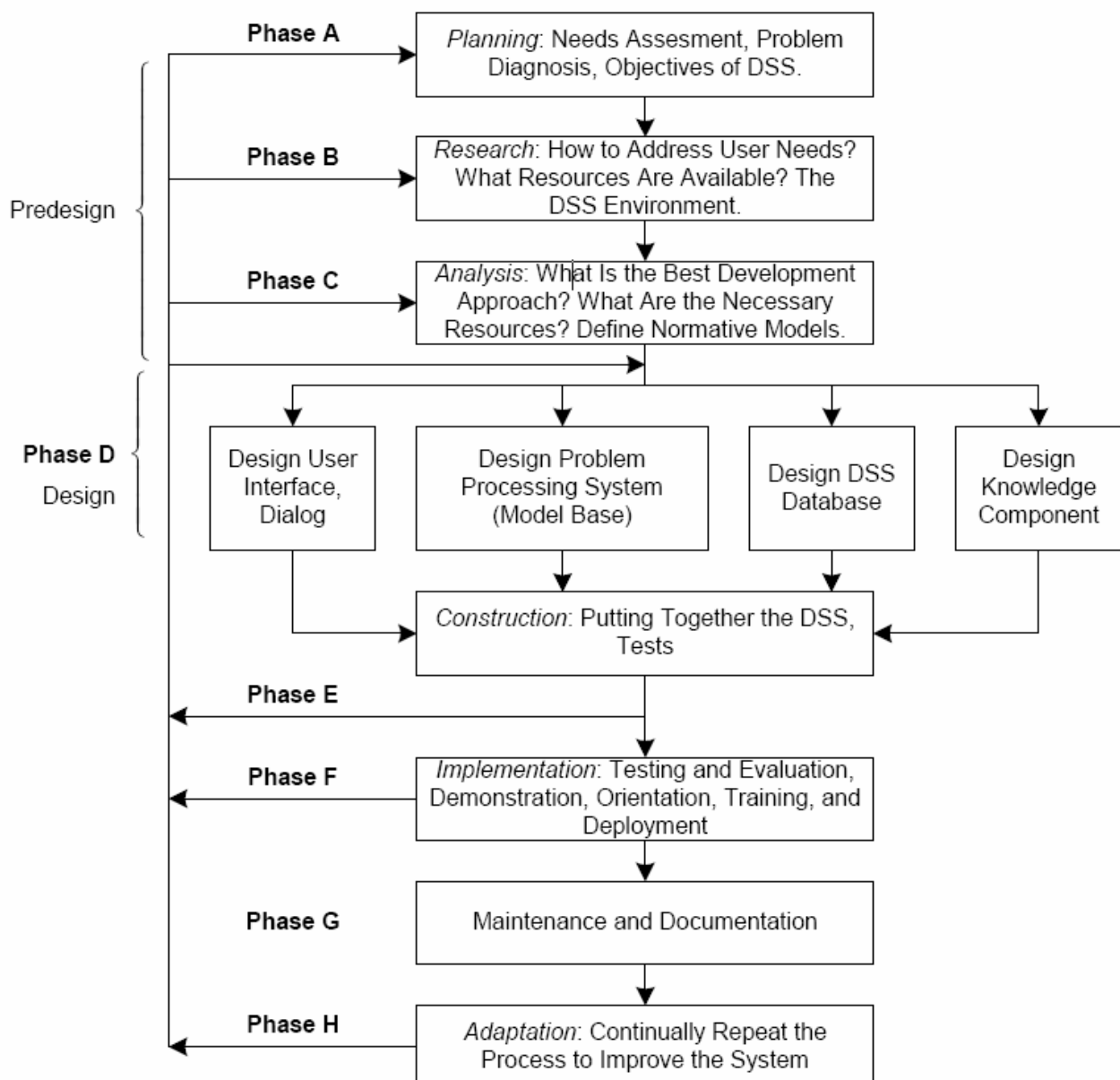
- Membangun sebuah DSS, apalagi yang besar, merupakan proses yang rumit.
- Melibatkan hal-hal: teknis (hardware, jaringan) dan perilaku (interaksi manusia-mesin, dampak DSS pada individu).
- Agar lebih mudah membangun DSS bisa digunakan bahasa khusus (misal CASE TOOLS).

Strategi Pengembangan.

1. Tulis DSS dengan bahasa pemrograman umum: Pascal, Delphi, C, C++, C#, Java, dan lainnya.
2. Menggunakan 4GL: data-oriented language, spreadsheets, dan financial-oriented language.
3. Menggunakan DSS Generator: Excel, Lotus 1-2-3, Quattro Pro, Express. Generator lebih efisien dari 4GL tapi ini tergantung juga pada batasannya.
4. Menggunakan DSS Generator khusus (domain specific): Commander FDC untuk budgeting & financial analysis, EFPM untuk kalangan perguruan tinggi.
5. Mengembangkan DSS dengan metodologi CASE. Memiliki jaminan kualitas yang memadai.
6. Untuk DSS yang kompleks, bisa mengintegrasikan pendekatan-pendekatan di atas.

Proses Pengembangan DSS.

- Pengembangan suatu DSS terkait juga dengan struktur permasalahan: tak terstruktur, semi terstruktur, ataupun terstruktur. Berturut-turut ES/DSS, DSS, MIS bisa digunakan untuk ini.
- Di bawah ini disajikan bagan fase-fase pengembangan suatu DSS:



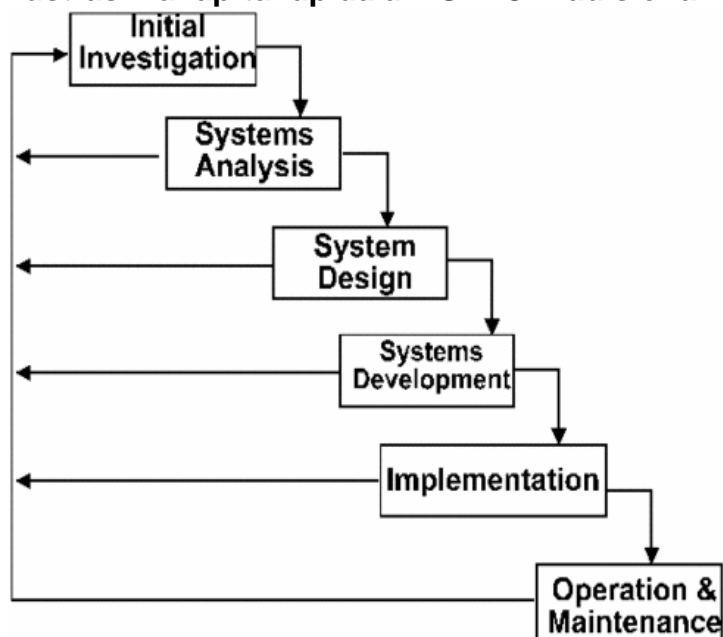
Proses Pengembangan: Life Cycle vs Prototyping.

Pembangunan DSS dilakukan dalam berbagai cara. Dibedakan antara pendekatan life cycle (daur hidup) dan iterative process (proses berulang).

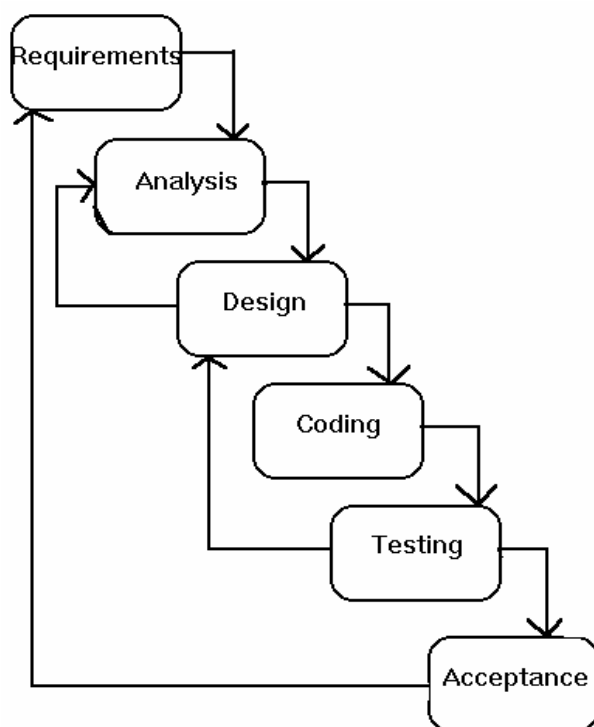
Pendekatan SDLC (System Development Life Cycle).

- Asumsi dasarnya adalah kebutuhan informasi dari suatu sistem dapat ditentukan sebelumnya.
- IRD (Information Requirements Definition) adalah pendekatan formal yang digunakan oleh sistem analis.
- IRD secara tradisional merupakan kombinasi analisis logik dengan pengamatan perilaku pemrosesan informasi.
- IRD bisa juga melibatkan CSF (Critical Success Factors).
- DSS didesain untuk membantu pengambilan keputusan para manajer pada masalah yang jelek strukturnya. Di satu sisi memahami kebutuhan user adalah hal yang sulit. Sehingga perlu diterapkan adanya bagian pembelajaran dalam desain atau proses kita. Dari situ, diharapkan user belajar mengenai masalah atau lingkungannya sehingga dapat mengidentifikasi kebutuhan informasi baru dan yang tak diantisipasi sebelumnya.

Ilustrasi Tahap-tahap dalam SDLC Tradisional



http://exonous.typepad.com/mis/2004/02/systems_develop.html



The waterfall model (Systems Development Life Cycle)

<http://www.albany.edu/acc/courses/fall97/acc681/ch7.html>

Pendekatan Prototyping Evolusioner.

- Pendekatan prototyping disebut juga proses evolusioner (evolutionary process), proses berulang (iterative process), atau cukup disebut prototyping saja. Nama lainnya adalah middle-out process (proses sementara), adaptive design (desain adaptif) dan incremental design (desain berkelanjutan).
- Proses desain berulang ini mengkombinasikan 4 fase utama SDLC tradisional (analisis, desain, konstruksi, dan implementasi) ke dalam 1 langkah yang diulang-ulang.
- Proses berulang terdiri dari 4 tugas, seperti di bawah ini:
 1. Memilih submasalah penting yang akan dibangun pertama kali.
 2. Mengembangkan sistem yang kecil, tapi berguna, dalam membantu pengambil keputusan.
 3. Mengevaluasi sistem terus menerus.
 4. Menghaluskan, mengembangkan, dan memodifikasi sistem secara berulang.

Pengembangan DSS Berbasis Tim dan Berbasis User.

- Pengembangan DSS pada tahun 1970 dan 1980-an melibatkan skala yang besar, sistemnya kompleks, dan didesain utama untuk mendukung organisasi. Sistem ini didesain oleh tim yang terdiri dari user, penghubung (intermediaries), DSS builder, tenaga ahli, dan berbagai tool. Berbagai individu dalam setiap kategori tadi, sehingga ukuran tim menjadi besar dan komposisinya sering berubah seiring berjalannya waktu. Intinya dengan berbasis tim, maka pembangunan DSS menjadi kompleks, lama, dan prosesnya memakan biaya.
- Pendekatan lainnya adalah membangun DSS berbasiskan user. Dimulai mulai tahun 1980-an, seiring pesatnya perkembangan di bidang PC (Personal Computer), jaringan komunikasi komputer, berkurangnya biaya hardware maupun software.
- Enterprise-wide computing serta kemudahan akses data dan pemodelan berarsitektur client/server juga mendukung pengembangan DSS berbasis user.
- Tentu saja kedua pendekatan ini bisa dikombinasikan, untuk mendapat kinerja yang diinginkan.

Pengembangan DSS Berbasis Tim.

Menentukan DSS Group.

Secara organisasi penempatan DSS Group bisa dimana-mana, umumnya pada lokasi:

1. Dalam departemen IS (Information Services).
2. Executive Staff Group.
3. Dalam wilayah keuangan atau fungsi lainnya.
4. Dalam departemen rekayasa industri.
5. Dalam kelompok manajemen pengetahuan (Management Science Group).
6. Dalam kelompok pusat informasi (Information Center Group).

Komputasi End-User dan Pengembangan DSS Berbasis User.

Komputasi End-User.

Pengembangan DSS berbasis user berelasi secara langsung kepada komputasi end-user. Definisinya adalah: pengembangan dan penggunaan sistem informasi berbasis komputer oleh orang-orang di luar wilayah sistem informasi formal. Definisi ini melibatkan juga manajer dan profesional yang menggunakan komputer pribadi, pengolah kata yang digunakan oleh sekretaris, e-mail yang digunakan oleh CEO, dan sistem time-sharing yang digunakan oleh ilmuwan dan peneliti.

Pengembangan DSS Berbasis User: Keuntungan dan Resikonya.

Berbagai keuntungan yang bisa didapat user bila dia sendiri yang membangun DSS:

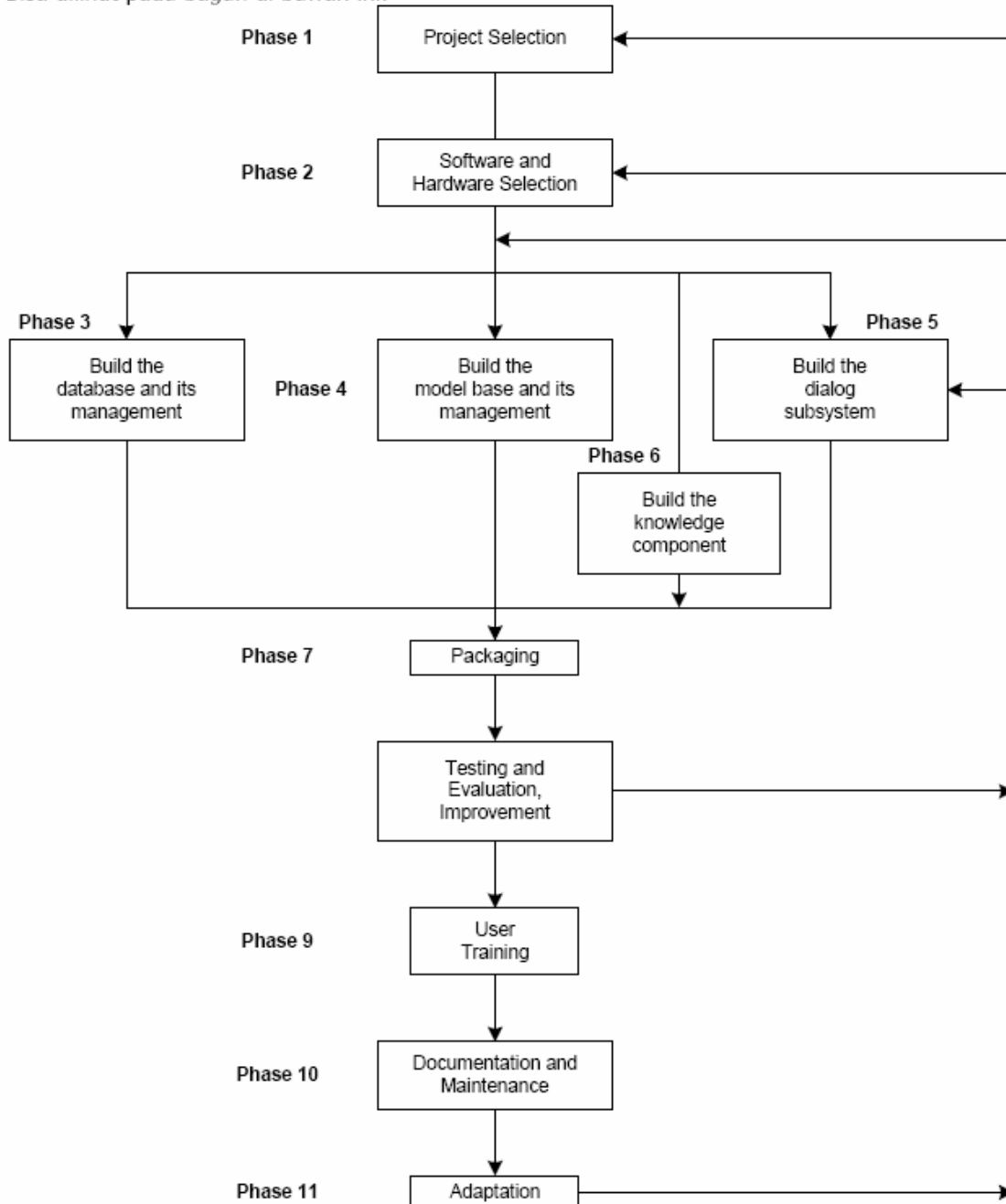
1. Waktu penyelesaiannya singkat.
2. Syarat-syarat spesifikasi kebutuhan sistem tak diperlukan.
3. Masalah implementasi DSS dapat dikurangi.
4. Biayanya sangat rendah.

Resikonya adalah:

1. Kualitasnya bisa tak terjaga.
2. Resiko potensial kualitas dapat diklasifikasikan dalam 3 kategori: (a) tool dan fasilitas di bawah standar, (b) resiko yang berhubungan dengan proses pengembangan (contoh: ketidakmampuan mengembangkan sistem yang bisa bekerja, pengembangan sistem yang menghasilkan hasil yang salah), dan (c) resiko manajemen data (misal: kehilangan data).

Proses Pembuatan.

Bisa dilihat pada bagan di bawah ini:



DSS Generator.

DSS Generator mengkombinasikan kemampuan berbagai aplikasi umum dalam 1 program.

Di bawah ini adalah program-program yang merupakan “bahan baku” bagi paket terintegrasi:

- Spreadsheet.
- Manajemen Data.
- Pengolah kata.
- Komunikasi.
- Grafis bisnis.
- Kalender (manajemen waktu).
- Desk management.
- Manajemen proyek.

Contoh dari paket terintegrasi ini adalah: Lotus 1-2-3, Microsoft Excel.

Pemilihan DSS Generator dan Tool Software Lainnya.

Berbagai pertanyaan yang harus dijawab oleh suatu organisasi yang akan menggunakan DSS Generator: (1) generator seperti apa yang akan digunakan, (2) hardware seperti apakah yang dipakai untuk menjalankannya, (3) sistem operasi seperti apa yang akan digunakan, (4) jaringan seperti apakah yang akan dipakai untuk menjalankannya.

Dengan kemampuan PC yang luar biasa sekarang ini, software DSS lebih banyak ditemui pada jenis komputer mikro. Kemudian dengan adanya program-program berbasis Windows, membuat DSS menjadi lebih disukai karena kemudahan penggunaannya.

Pemilihan Software.

Tool software dasar yang patut dipertimbangkan adalah:

- Fasilitas database relasional dengan fasilitas pembuatan laporan yang baik dan fasilitas pemilihan data setiap saat.
- Bahasa penghasil grafis.
- Bahasa pemodelan.
- Bahasa analisis data statistik umum.
- Bahasa khusus yang lain (misal: untuk membangun simulasi).
- Bahasa pemrograman (generasi ketiga).
- Tool pemrograman berorientasi objek.
- Tool pembangun ES.
- Jaringan.
- CASE tools.

Kesimpulan.

- DSS dikembangkan dengan proses pengembangan yang unik berdasarkan prototyping.
- Langkah utamanya adalah: perencanaan, riset, analisis, desain, konstruksi, implementasi, perawatan, dan adaptasi.
- Pendekatan berulang (prototyping) paling umum digunakan dalam DSS, karena kebutuhan informasi tak dapat diketahui dengan tepat pada awal proses.
- DSS dapat dibangun oleh tim maupun individu.
- Pembangunan DSS dengan tim mengikuti proses terstruktur, termasuk perencanaan, pemilihan software yang sesuai (generator jika dibutuhkan), dan hardware.
- Bagian utama dari komputasi end-user adalah pembangunan DSS untuk dukungan personal yang dilakukan oleh individu.
- Keuntungan utama orang-orang yang membangun DSS-nya sendiri adalah: waktu penyelesaiannya singkat, familiar dengan kebutuhannya, biaya rendah, dan implementasinya lebih mudah.
- Pengembangan DSS berbasis user bisa juga berkualitas rendah, karena itu kontrol yang cukup dapat memperbaiki situasi tersebut.
- Kebanyakan DSS dibangun dengan generator pengembangan DSS atau dengan tool-tool pengembangan 4GL tak terintegrasi.
- Terdapat banyak sekali tool dan generator di pasaran. Pemilihan yang sesuai untuk membangun DSS tertentu haruslah didesain dengan cermat.
- Banyak DSS dibangun dalam lingkungan Windows. Windows membuatnya mungkin untuk membangun DSS dengan cepat dan murah.

SISTEM INFORMASI PERUSAHAAN

KONSEP DAN DEFINISI

SIP → Merupakan sebuah sistem yang dapat digunakan oleh para Eksekutif dan semua karyawan.

Tujuan → memberikan informasi holistik dari sudut pandang perusahaan sehingga dapat dijadikan sebagai manajemen sumber daya perusahaan

SIE → merupakan sebuah sistem yang hanya dapat digunakan oleh para Eksekutif yang sifatnya dapat melakukan *drill down*.

Tujuan → melayani kebutuhan informasi bagi para eksekutif puncak

Teknologi SIP dan dampak dari Teknologi Web

Teknologi SIP	Web	Dampak
SIE	Konsisten, userfriendly, GUI, cepat dan akses langsung terkoneksi ke Data Base/data warehouse	Manajemen yang lebih baik dan penggunaan sumber web pada tingkat grafis
SIP	s.d.a	s.d.a

EVOLUSI SIE DAN SIP

SIE=SPK → Hal ini dikarenakan para eksekutif tidak cukup hanya dengan adanya aplikasi SPK, tetapi juga adanya suatu pusat data/informasi dengan kriteria data yang dibutuhkan :

- 1) menyediakan data dan informasi (50%)
- 2) untuk penjadwalan (50%)
- 3) pengarahan singkat elektronik (50%)
- 4) Untuk browsing internet dan situasi monitoring (31,3%)

Kemajuan data warehouse dan web yang meningkat

Mengapa SIE..?

Karena mempunyai manfaat : dapat meningkatkan kuantitas dan kualitas informasi yang tersedia bagi para eksekutif

Kebutuhan Informasi :

- a. informasi yang tepat waktu
- b. akses lebih besar ke data operasional
- c. informasi relevan yang lebih singkat
- d. informasi baru
- e. informasi lebih banyak tentang lingkungan eksternal
- f. informasi yang lebih kompetitif
- g. akses lebih cepat ke database eksternal
- h. akses lebih cepat ke informasi
- i. mengurangi biaya kertas

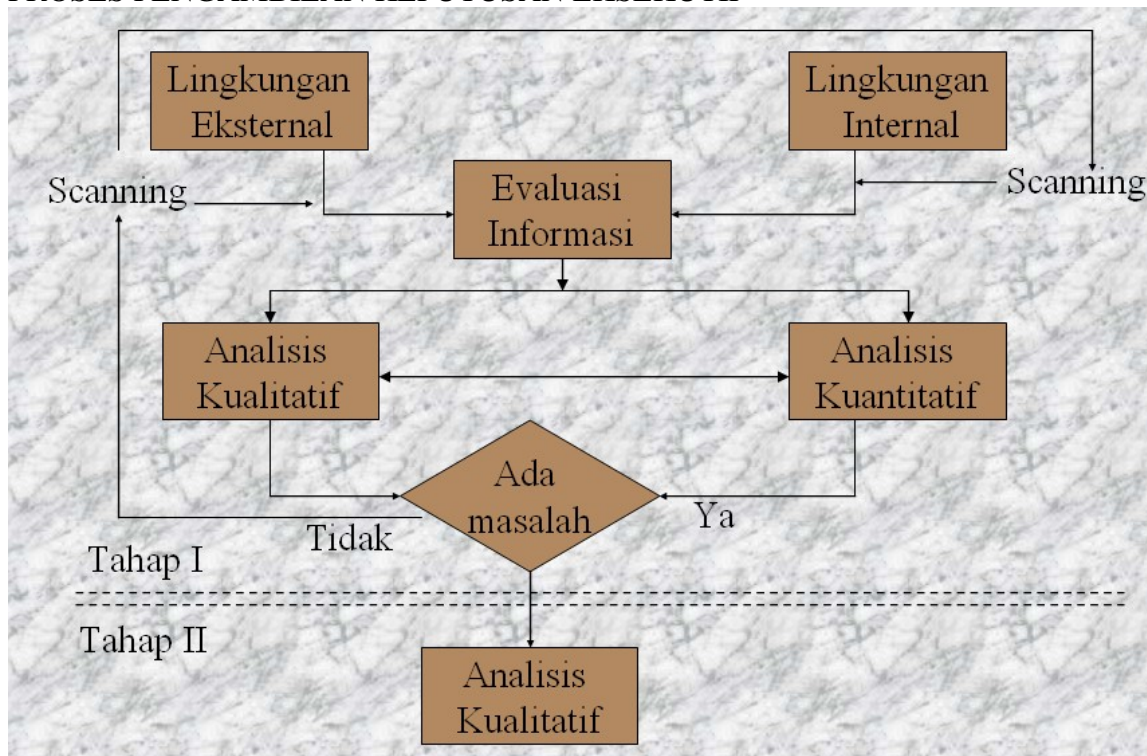
Meningkatkan kinerja eksekutif :

- a. meningkatkan komunikasi
- b. kemampuan lebih besar untuk mengidentifikasi tren historis
- c. meningkatkan efisiensi dan efektivitas eksekutif
- d. perhatian lebih fokus
- e. meningkatkan rentang pengawasan
- f. dukungan lebih besar untuk pengambilan keputusan eksekutif

SISTEM PENDUKUNG PERUSAHAAN

- DEFINISI : merupakan sebuah sistem yang dapat digunakan/dijadikan alat untuk mendukung perusahaan seperti : OA dan dukungan analisis.

PROSES PENGAMBILAN KEPUTUSAN EKSEKUTIF



KARAKTERISTIK DAN MANFAAT SIE

Kriteria	Deskriptif
<ul style="list-style-type: none"> ■ Kualitas Informasi ■ Antar Muka Pengguna 	<ul style="list-style-type: none"> ■ Fleksibel ■ Menghasilkan informasi yang relevan, lengkap dan valid ■ Menghasilkan informasi tepat waktu dan benar ■ Memasukkan antar muka yang ramah pengguna ■ Dapat diakses dari banyak tempat ■ Disesuaikan dengan style manajemen untuk masing-masing eksekutif
<ul style="list-style-type: none"> ■ Manfaat ■ Menyediakan kapabilitas teknis 	<ul style="list-style-type: none"> ■ Memfasilitasi pencapaian sasaran organisasi ■ Memfasilitasi akses ke informasi ■ Memberikan keuntungan kompetitif ■ Menghemat waktu pengguna ■ Meningkatkan kapasitas komunikasi dan kualitasnya ■ Interpretasi tertulis ■ Akses ke email ■ Memberikan forecasting ■ Informasi dihasilkan ke berbagai tingkat detail

MANAJEMEN PENGETAHUAN

Pengantar

- Pengetahuan dipandang sebagai sebuah komoditi atau sebuah aset intelektual
- Karakteristik pengetahuan:
 1. Penggunaan pengetahuan tidak akan menghabiskannya
 2. Perpindahan pengetahuan tidak akan menghilangkannya
 3. Pengetahuan itu berlimpah, tetapi kemampuan terbatas untuk menggunakannya
 4. Banyak pengetahuan berharga hilang begitu saja

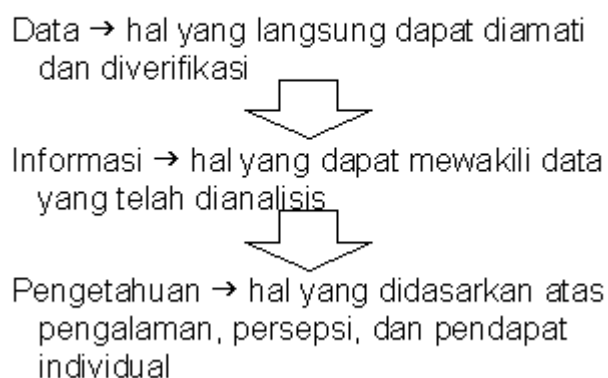
Definisi Awal Manajemen Pengetahuan

Proses penerapan pendekatan sistematis untuk menangkap, menyusun, mengelola, dan menyebarkan pengetahuan di seluruh organisasi untuk mempercepat kerja, menggunakan best practice, dan mengurangi biaya pengerjaan ulang dari proyek ke proyek

Transformasi Manajemen Pengetahuan

- Manajemen Pengetahuan sudah lama diimplementasikan jauh sebelum istilah Manajemen Pengetahuan (Knowledge Management) diperkenalkan
- 1969 ARPANET sebagai cikal bakal jaringan internet
- Fase-fase perkembangan meliputi:
 - 1800 → industrialisasi
 - 1850 → transportasi
 - 1900 → komunikasi
 - 1950 → komputerisasi
 - 1980 → visualisasi
 - 2000 → personalisasi
- Era Industri → Era Pengetahuan
- Sedikit orang mengerjakan lebih banyak pekerjaan
- Di Era Pengetahuan, organisasi harus dapat belajar, mengingat, dan bertindak berdasarkan informasi, pengetahuan, *know-how* yang tersedia
- Memiliki dasar pengetahuan (*knowledge base*) → *lessons learned* dan *best practice* yang berharga

Dari Data Menjadi Pengetahuan



Jenis dan Bentuk Pengetahuan

1. *Tacit knowledge*

- Pengetahuan yang sulit diartikulasikan, dituliskan dalam kata-kata, teks, maupun gambar
- Tersimpan dalam pikiran manusia, sulit diformulasikan (misalnya keahlian seseorang)
- Penting untuk kreatifitas dan inovasi
- Dikonversikan ke eksplisit dengan eksternalisasi
- Misalnya pengalaman bertahun-tahun yang dimiliki oleh ahli

2. *Explicit knowledge*

- Pengetahuan yang telah ditangkap dan dinyatakan dalam kata-kata, teks, maupun gambar
- Telah ada dalam bentuk konkrit/nyata
- Dapat dikodifikasi/formulasi
- Dikonversikan ke *tacit* dengan pemahaman dan penyerapan
- Misalnya dokumen, database, materi audio visual dll

→ Semakin tacit sebuah pengetahuan, maka semakin berharga pengetahuan tersebut

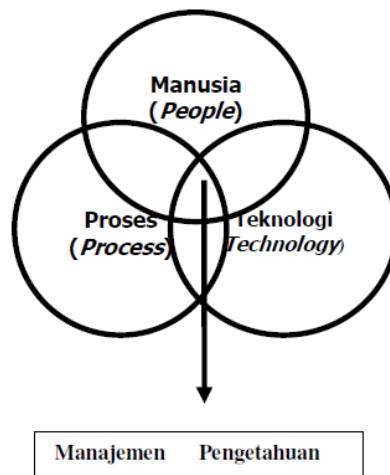
Pendekatan Pengembangan Manajemen Pengetahuan

Manajemen pengetahuan bukan perkara yang sederhana, karena luas dan kompleksnya bidang manajemen pengetahuan ini para ahli mencoba membangun model untuk manajemen pengetahuan. Manajemen Pengetahuan dilaksanakan dalam sistem pengelolaan pengetahuan, atau *Knowledge Management System* (KMS). Sebagian besar organisasi yang menerapkan KMS, menggunakan pendekatan tiga-cabang untuk mengelola pengetahuannya, yaitu

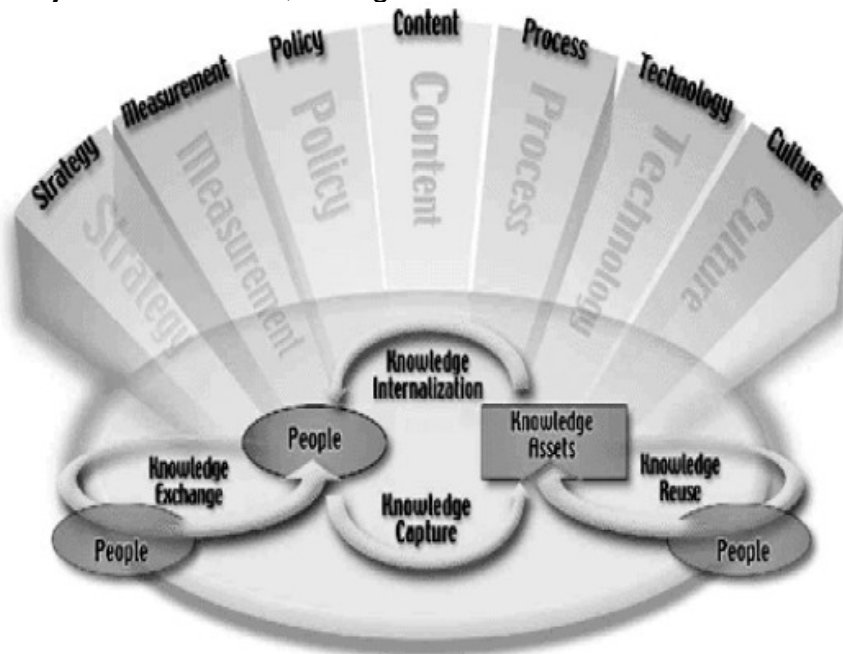
- Manusia (*People*),
- Proses (*Process*), dan
- Teknologi (*Technology*).

Penekanan terhadap tiap-tiap elemen bisa berbeda di setiap bagian organisasi.

Berdasarkan model pendekatan di atas maka batasan dari *knowledge management* sebagai berikut :



Model lain adalah yang dikemukakan oleh ahli lain yang membagi model manajemen pengetahuan menjadi dua dimensi, sebagai berikut:



The EDGE, April 2000, Vol. 4 No. 1.

Dimensi pertama (bawah) terdiri dari aktifitas-aktifitas yang sangat penting bagi proses penciptaan pengetahuan dan inovasi seperti :

- *knowledge exchange*,
- *knowledge capture*,
- *knowledge reuse*, dan
- *knowledge internalization*.

Secara keseluruhan, proses ini menciptakan sebuah organisasi pembelajaran (*learning organization*) yaitu sebuah organisasi yang memiliki keahlian dalam penciptaan, perolehan, dan penyebaran pengetahuan serta mengadaptasikan aktifitasnya untuk merefleksikan pemahaman dan inovasi baru yang didapat.

Sedangkan dimensi kedua (atas) terdiri dari elemen yang memungkinkan atau mempengaruhi aktifitas penciptaan pengetahuan, yaitu:

- *Strategy* – penyelarasan strategi organisasi dengan strategi KMS.
- *Measurement* – pengukuran yang diambil untuk menentukan apakah terjadi perbaikan KM atau ada manfaat yang telah diambil.
- *Policy* – aturan tertulis atau petunjuk-petunjuk yang telah dibuat oleh organisasi.
- *Content* – bagian dari knowledge-base organisasi yang ditangkap secara elektronik.
- *Process* – proses-proses yang digunakan oleh *knowledge worker* organisasi dalam rangka mencapai misi dan tujuan organisasi.
- *Technology* – teknologi informasi yang memfasilitasi proses identifikasi, penciptaan, dan difusi pengetahuan diantara elemen-elemen organisasi di seluruh bagian organisasi. Peran penting teknologi dalam KMS adalah memperluas jangkauan dan meningkatkan kecepatan transfer pengetahuan. Peran ini sangat tergantung pada dua aspek yang paling banyak mendukung, yaitu penyimpanan dan komunikasi.
- *Culture* – lingkungan dan konteks yang di dalamnya proses-proses KM harus terjadi (sering disebut dengan istilah nilai, norma, dan praktek).

Aplikasi Teknologi Informasi Pada Manajemen Pengetahuan

Perkembangan teknologi informasi memainkan peranan amat penting dalam perkembangan konsep manajemen pengetahuan. Dalam catatan Beckman (1999, h.1.2), peristiwa penting yang menandai tonggak perkembangan manajemen pengetahuan adalah ketika di tahun 1980 organisasi DEC (*Digital Equipment Corporation*) dan Universitas Carnegie mellon mengembangkan sistem pakar untuk menetapkan konfigurasi perangkat keras komputer. Sejak itu banyak penelitian yang menuju pada pemanfaatan teknologi untuk memanfaatkan pengetahuan yang tersimpan di kepala manusia. Namun baru enam tahun kemudian istilah “manajemen pengetahuan” diperkenalkan secara formal oleh Dr. Karl Wiig dalam sebuah pidatonya di konferensi ILO (badan buruh PBB).

Seperti yang telah dijelaskan sebelumnya, bahwa dalam penerapan manajemen pengetahuan dapat didukung dengan teknologi informasi. Oleh karena itu, komponen selanjutnya dalam penerapan manajemen pengetahuan ini adalah teknologi; dalam hal ini berkaitan dengan pemanfaatan Teknologi Informasi (TI). Istilah Teknologi Informasi merupakan gabungan dua istilah dasar yaitu teknologi dan informasi. Teknologi dapat diartikan sebagai pelaksanaan ilmu, sinonim dengan ilmu terapan. Sedangkan, secara umum informasi merupakan sesuatu arti yang diungkapkan oleh manusia atau oleh ekstrak dari fakta.

Teknologi tidak saja terbatas pada perangkat keras (alat) dan perangkat lunak (program), tetapi juga mengikutsertakan manusia serta tujuan yang ditentukan, nilai yang digunakan untuk membuat pilihan pelaksanaan, dan kriteria penilaian yang digunakan untuk memutuskan apakah manusia mengendalikan teknologi atau tidak. Yang termasuk teknologi informasi adalah antara lain:

- (1) telekomunikasi,
- (2) sistem komunikasi optik,
- (3) sistem pita-video dan cakram video,
- (4) komputer, termasuk visi komputer, lingkungan data dan sistem pakar,
- (5) mikrobentuk,
- (6) komunikasi suara dengan bantuan komputer,
- (7) jaringan kerja data,
- (8) surat elektronik, dan (videoteks dan teleteks.)

Keberadaan teknologi informasi mampu menawarkan berbagai metode, antara lain :

1. Metode dan tools untuk merekam pengetahuan termasuk komputer, media simpan seperti pita magnetis dan cakram atau disc. Pencarian data teks lengkap memungkinkan pemakai menelusuri direktori, ensiklopedia, data statistik, dan keuangan yang terbacakan mesin. Ini semua dipermudah dengan tersedianya media simpan optik.
2. Metode menyimpan rekaman (*record*) mengenai berbagai kegiatan termasuk perangkat keras komputer seperti media simpan, yang dilengkapi perangkat lunak untuk merancang bangun, menciptakan, dan menyunting data, spreadsheet, dan perangkat lunak sejenis.
3. Metode untuk mengindeks dokumen dan informasi termasuk berbagai teknik pembuatan indeks berbantuan komputer serta berkas (*files*) khusus untuk memudahkan menemukan dokumen berdasarkan istilah atau kondisi istilah dalam berkas. Pangkalan data bibliografis yang besar yang memudahkan menemukan dokumen yang memenuhi syarat tertentu (misalnya berdasarkan pengarang atau subjek), kini berkembang dengan katalog sehingga membantu menentukan lokasi dokumen.
4. Metode mengkomunikasikan pengetahuan termasuk :
 - a. sistem pos elektronik untuk transmisi teks memo dan surat dokumen ;
 - b. system transmisi faksimili (*facsimile*) untuk transmisi dokumen jarak jauh berdasarkan prinsip fotokopi. Ini sama saja dengan fotokopi jarak jauh ;
 - c. majalah elektronik sebagai sarana komunikasi kegiatan dan hasil penelitian ;
 - d. telekonferensi artinya pertemuan jarak jauh, masing-masing peserta berada di berbagai tempat, saling berkomunikasi serta terlihat wajah masing-masing ; dan
 - e. jaringan komunikasi data untuk mengkomunikasikan data.

Sementara itu, ketika teknologi jaringan dan telekomunikasi semakin maju, maka boleh dikatakan bahwa teknologi pengelolaan pengetahuan mengalami pertumbuhan sangat dinamik. Seperti dikatakan Jablonski, Horn, dan Schlundt (2001), manajemen pengetahuan kini berdiri di atas tiga kaki yaitu :

- Intelegensi buatan (*artificial intelligence*) yang membantu mengekstraksi informasi dari berbagai sumber untuk disimpan di *knowledge base*. Sebuah *knowledge base* memiliki format yang bisa ditelusur dan diakses sesuai keperluan pemakai. Pendekatannya berdasarkan asumsi bahwa *knowledge base* bisa dipisahkan dari *knowledge carriers*.
- Manajemen dokumen (*document management*) untuk menyimpan dan mengelola berbagai tipe dokumen di dalam satu pusat. Pemanfaatannya adalah melalui metadata.
- Teknologi jaringan komputer dan *hypertext* yang memungkinkan berbagai dokumen dihubungkan, sedangkan pencariannya didukung oleh *search engine*.

Majunya Teknologi Informasi (TI) memang bisa memacu efisiensi dan efektifitas organisasi. Karena dirasa banyak manfaatnya bagi organisasi, sehingga usaha-usaha untuk lebih memaksimalkan TI terus berkembang. Bagi mereka, TI telah menjadi bagian yang tak terpisahkan dan merupakan infrastruktur yang penting bagi organisasi atau organisasi dalam memberikan nilai tambah atau keuntungan kompetitif.

Multidisiplin manajemen pengetahuan

- Ilmu keorganisasian
- Ilmu kognitif
- Linguistik
- Teknologi informasi → *knowledge-based system, database technology, information management*
- Ilmu kepustakaan
- Teknik penulisan dan jurnalisme
- Antropologi dan sosiologi
- Pendidikan dan pelatihan
- Ilmu komunikasi
- Teknologi kolaborasi → intranet, ekstranet, portal, *web technologies*