

1. PENGANTAR SIMULASI

Pengertian

“Simulasi adalah imitasi atau tiruan dari aktivitas/proses sebuah system”



“Simulasi dibuat dengan tujuan untuk mengamati karakteristik sistem nyatanya”

“Terkadang simulasi dibuat untuk sesuatu yang belum ada sistem nyatanya, sehingga pembuatan simulasi dalam hal ini adalah untuk menguji sistem rancangan”



“Simulasi bisa tidak menggunakan komputer, tetapi cukup menggunakan persamaan-persamaan matematik”

“Namun demikian, kompleksitas sistem nyata biasanya mengharuskan penggunaan komputer agar model yang disimulasikan mendekati atau sangat mirip dengan sistem nyata”



Beberapa Definisi

- *“Simulation is a situation in which a particular set of conditions is created artificially in order to study or experience sth that could exist in reality” (The Oxford Business English Dictionary, 2010).*
- *“Simulation is “the modeling of a process or system in such a way that the model mimics the response of the actual system to events that take place over time” (Schriber,1987).*
- *“Simulation is the imitation of the operation of a real world process or system over time” (Banks at al, 2001).*



“Simulasi banyak digunakan saat ini sebagai suatu tool studi.”

- ❖ simulasi dapat digunakan sebagai pembelajaran/investigasi terhadap interaksi antar komponen dalam suatu sistem yang kompleks
- ❖ simulasi dapat digunakan untuk studi dampak perubahan informasi, organisasi, dan lingkungan terhadap sistem
- ❖ pengetahuan yang diperoleh melalui simulasi bisa menjadi saran penting untuk perbaikan sistem, mengidentifikasi variabel yang paling berpengaruh terhadap sistem melalui pengubahan input terhadap output.



“Simulasi juga sebagai media untuk memahami metodologi penyelesaian masalah yang dilakukan secara analitik, alat percobaan untuk desain sistem baru, analisis kemampuan mesin, dan lain sebagainya”.

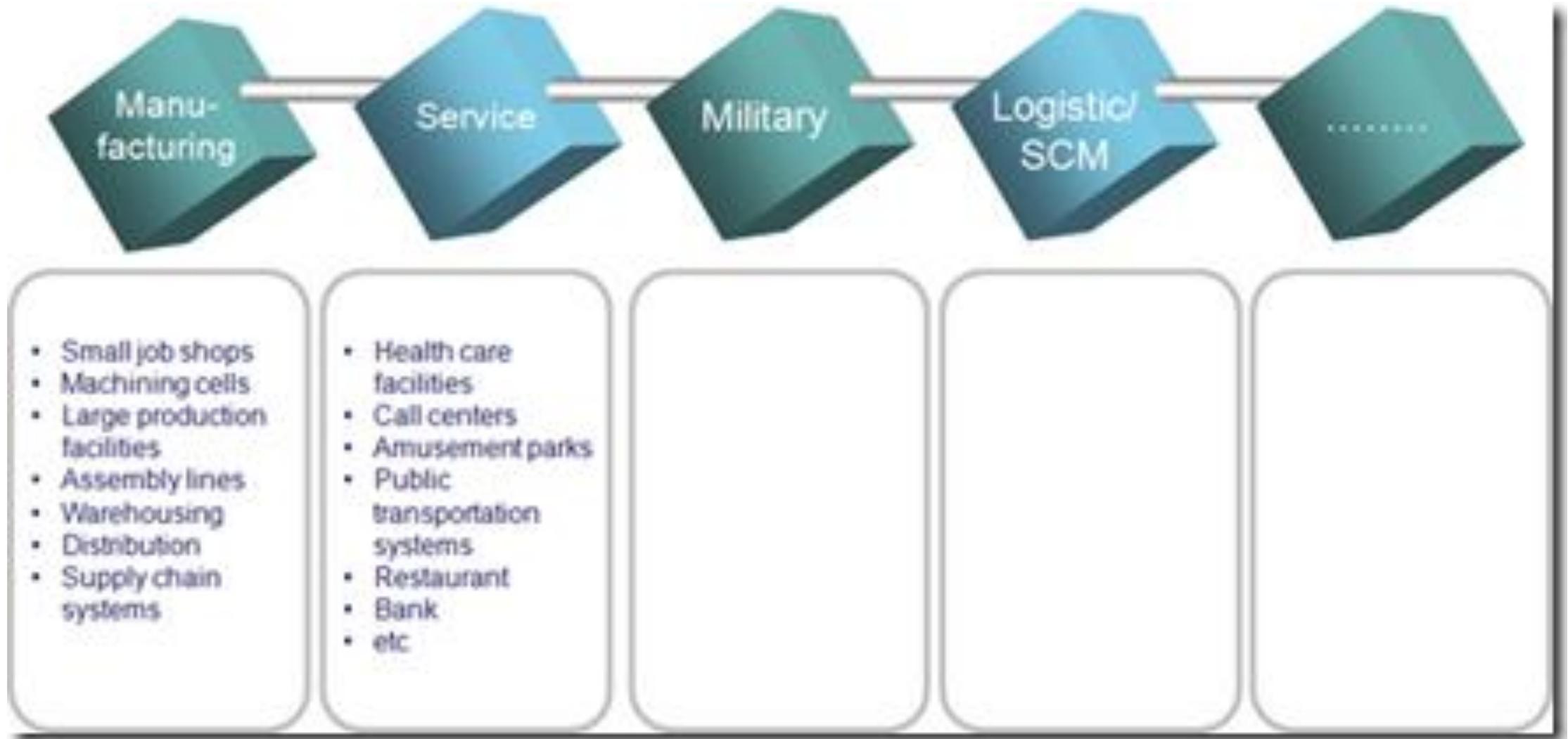


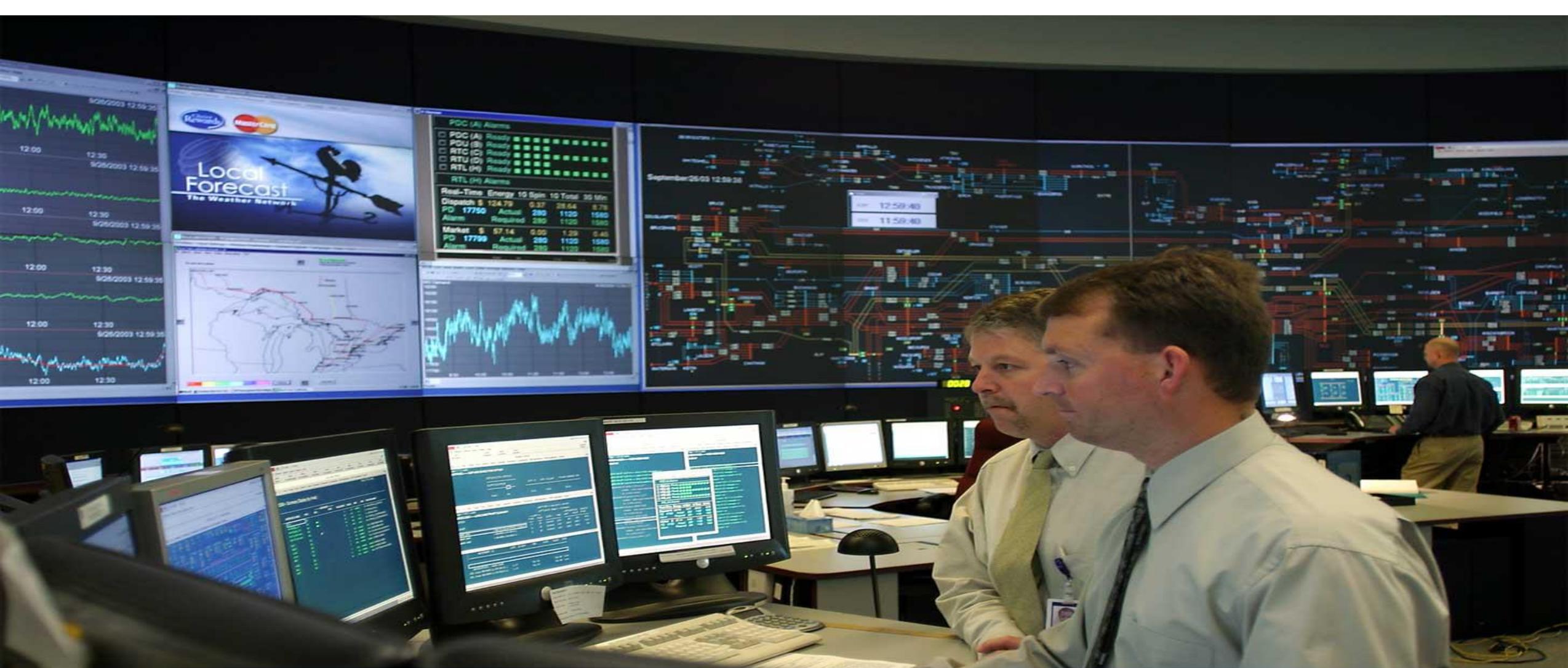
Simulasi tidak tepat digunakan untuk suatu studi apabila hal-hal berikut ini ditemui:

1. Suatu masalah dapat diselesaikan dengan nalar.
2. Suatu masalah dapat diselesaikan secara analitik.
3. Suatu percobaan sederhana dapat dilakukan.
4. Biaya simulasi sangat besar.
5. Waktu dan sumber daya yang tidak tersedia.
6. Tidak tersedia data (termasuk data estimasi/hipotetik).
7. Adanya ekspektasi yang berlebihan terhadap hasil.
8. Sistem terlalu kompleks dan tidak bisa didefinisikan.



Area Penggunaan Simulasi





2.KONSEP DASAR & MODEL SIMULASI

Pengertian

“Sistem adalah keseluruhan yang dinamis dan kompleks, berinteraksi sebagai suatu unit fungsional yang terstruktur”

“Sistem adalah keseluruhan yang dinamis dan kompleks, berinteraksi sebagai suatu unit fungsional yang terstruktur”

“Sistem seringkali tersusun atas entitas yang mencari keseimbangan, tetapi dapat memperlihatkan perilaku berisolasi, chaotic, atau eksponensial”



Menurut KBBI

sis.tem /sistém/

bentuk tidak baku: **sistim**

1. **n** perangkat unsur yang secara teratur saling berkaitan sehingga membentuk suatu totalitas: - *pencernaan makanan, pernapasan, dan peredaran darah dalam tubuh; -- telekomunikasi*
2. **n** susunan yang teratur dari pandangan, teori, asas, dan sebagainya: -- *pemerintahan negara (demokrasi, totaliter, parlementer, dan sebagainya)*
3. **n** metode: -- *pendidikan (klasikal, individual, dan sebagainya); kita bekerja dengan -- yang baik; -- dan pola permainan kesebelasan itu banyak mengalami perubahan*

DEFINISI

Kumpulan dari elemen-elemen yang bersama-sama berfungsi mencapai beberapa tujuan



Contoh Sistem

SISTEM
MANUFAKTUR

SISTEM
INFORMASI

SISTEM
PENCERNAAN



Komponen Sistem

ENTITAS

Entitas atau objek kajian dalam sistem adalah item-item objek yang diproses melalui sistem seperti produk, pelanggan, dan dokumen.

Entitas dapat memiliki karakteristik-karakteristik yang unik seperti biaya, ukuran, prioritas, kualitas, atau kondisi. Entitas dapat dikelompokkan menjadi manusia atau hewan (pelanggan, pasien, dsb), non-manusia (produk, komponen, dokumen, sampah, dsb) dan intangible (panggilan telepon, surat elektronik, dsb)

Atribut adalah properti atau sifat suatu entitas

ATRIBUT



Komponen Sistem

Aktivitas dapat diartikan sebagai kegiatan/tugas yang dilakukan entitas dalam periode waktu tertentu, dapat juga dipandang sebagai tugas-tugas yang dilakukan dalam sistem (baik secara langsung maupun tidak langsung) dalam memproses entitas.



ACTIVITIES

Contoh aktivitas adalah melayani nasabah, memotong komponen, memperbaiki sebuah peralatan, dsb. Aktivitas memerlukan waktu dan kadangkalan memerlukan sumber daya. Aktivitas dapat diklasifikasikan sebagai aktivitas memproses entitas (check-in, inspeksi produk, fabrikasi, dsb), perpindahan entitas dan sumber daya (pepindahan forklift, pepindahan dalam ban berjalan, etc), dan penyesuaian/perbaikan/perawatan sumber daya (setup mesin, perbaikan mesin, dsb



Komponen Sistem

STATE

State adalah variable yang mendeskripsikan status sistem pada satu waktu tertentu relatif terhadap tujuan kajian

EVENT

Event adalah suatu kejadian (instan) yang dapat mengubah status sistem (variable state). Endogeneous event adalah aktivitas atau kejadian yang terjadi dalam sistem, misalnya kejadian selesainya sebuah pelayanan seorang nasabah. Exogenous event adalah aktivitas atau kejadian di luar sistem (lingkungan) yang mempengaruhi sistem, misal kedatangan seorang nasabah ke dalam bank.



Komponen Sistem

RESOURCES

Sumber daya (resources) adalah suatu objek atau peralatan yang digunakan dalam melakukan suatu aktivitas. Sumber daya menyediakan dukungan fasilitas, peralatan, dan tenaga kerja untuk menyelesaikan suatu pekerjaan.

Sumber daya biasanya memiliki karakteristik berupa kapasitas, kecepatan, cycle time, dan reliability.

Sumber daya dapat dikelompokkan berupa human or animate (operator, dokter, petugas perbaikan, dsb.), inanimate (peralatan, perlengkapan, ruang kerja, dsb.), dan intangible (informasi, daya listrik, dsb.). Sumber daya juga dapat diklasifikasi sebagai sumber daya yang dedicated atau shared, permanent atau consumable, dan mobile atau stationary



DEFINISI MODEL

Menurut KBBI

mo.del¹ /modèl/

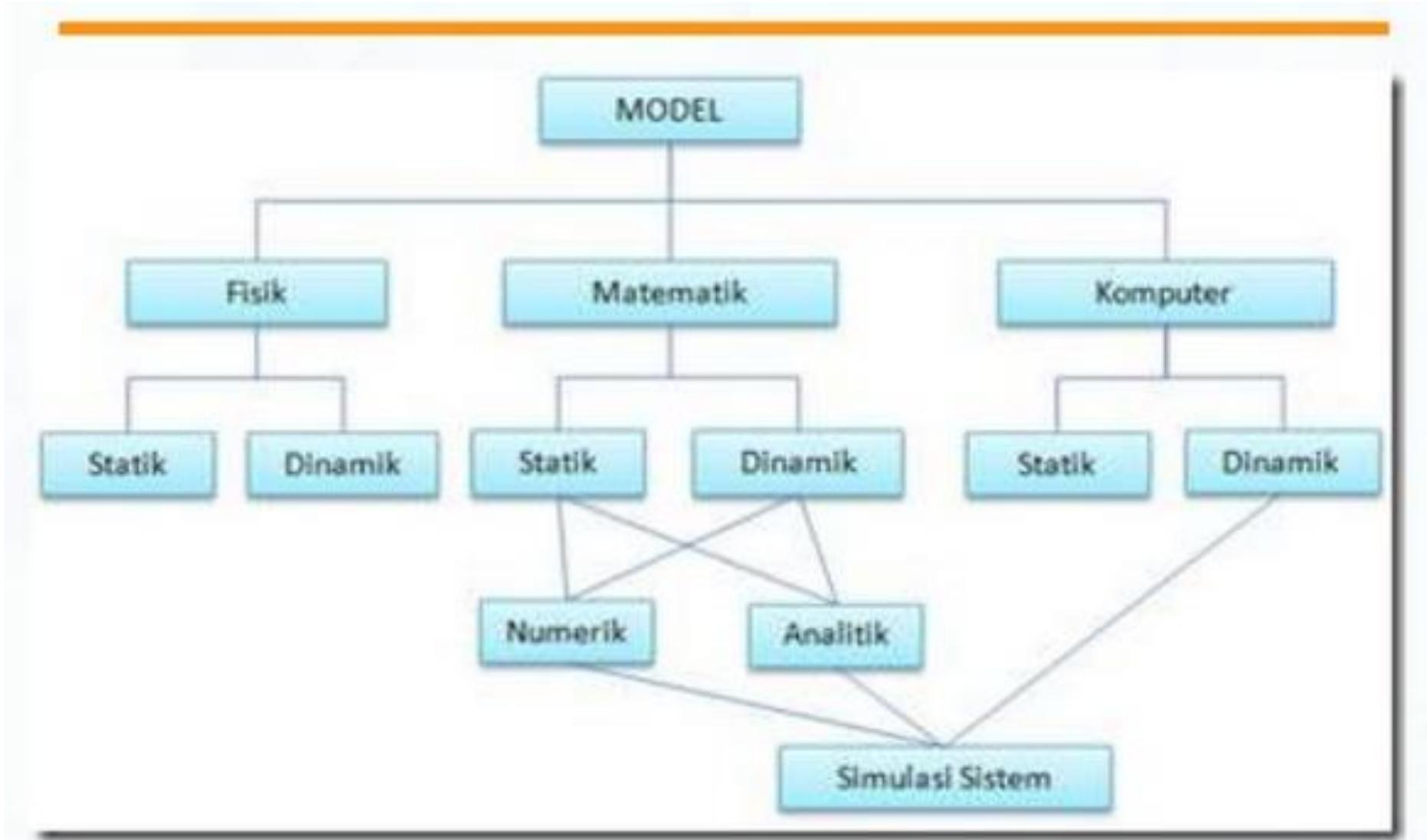
1. *n* pola (contoh, acuan, ragam, dan sebagainya) dari sesuatu yang akan dibuat atau dihasilkan: *rumahnya dibuat seperti -- rumah adat*
2. *n* orang yang dipakai sebagai contoh untuk dilukis (difoto): *pernah aku menjadi -- lukisan*
3. *n* orang yang (pekerjaannya) memperagakan contoh pakaian yang akan dipasarkan: *gadis -- yang cantik-cantik itu memperagakan pakaian dari bahan batik*
4. *n* barang tiruan yang kecil dengan bentuk (rupa) persis seperti yang ditiru: -- *pesawat terbang*

“Model adalah tiruan sebuah sistem yang disusun untuk mempelajari karakteristik sistem nyatanya.

Karakteristik sistem yang digambarkan dalam model biasanya tidak menyeluruh, melainkan disesuaikan dengan kebutuhan tujuan studi”



Klasifikasi Model Menurut Singh (2009)



Model Fisik

- ✓ Contoh model fisik statik adalah model bangunan yang dirancang oleh para arsitektur maupun teknik sipil. Model tersebut dapat berupa gambar maupun maket bangunan
- ✓ Contoh model fisik dinamik adalah model pesawat (berukuran kecil) yang sedang dalam pengujian di ruang pengujian angin. Ruang pengujian tersebut berupaya mencontoh kondisi udara, kecepatan dan lain sebagainya dengan berbagai kondisi ukuran untuk menguji model pesawat yang akan dibangun.
- ✓ Contoh lain model fisik dinamik adalah model bangunan anti erosi yang akan dibangun di pantai, atau model bangunan anti gempa



Model Matematik

- ✓ Model matematik merupakan imitasi sistem nyata dalam bentuk simbol-simbol matematik
- ✓ Model matematik statik tidak mempertimbangkan waktu dalam pengolahan datanya sehingga sistem tidak berubah oleh waktu, sedangkan model matematik dinamik adalah sebaliknya.
- ✓ Contoh model matematik adalah model inventori (persediaan). Model persediaan ini ada yang statik yaitu yang data permintaan, data lead time diasumsikan bersifat statik (deterministik), ada pula model persediaan dinamik dimana data permintaan, dan lead time bersifat probabilistik.



Model Komputer

Model komputer merupakan perkembangan lanjut dalam pemodelan karena seluruh model matematik baik statik maupun dinamik dapat dimodelkan secara lebih baik melalui komputer. Model komputer dinamik dapat kita lihat secara sederhana pada model permainan (game) yang meniru dunia nyata. (Singh, 2009)



Klasifikasi Model Menurut Banks (2001)

Berbeda dengan Singh (2009), Banks et al (2001) hanya mengelompokkan model dalam dua jenis yaitu model fisik dan model matematik. Model matematik ini dapat dinyatakan dalam bentuk **notasi simbol** atau **persamaan matematik**, dan bisa juga disajikan dalam bentuk **model simulasi**. Model simulasi ini kemudian lebih jauh dapat diklasifikasikan sebagai model simulasi statik atau dinamik, model simulasi deterministik atau stokastik, dan model simulasi diskrit atau kontinu.



Model Simulasi Statik

- Model simulasi statik dikenal juga dengan nama Simulasi Monte Carlo yang merepresentasikan sebuah sistem pada suatu waktu tertentu.
 - contoh, simulasi jumlah pelanggan yang membeli suatu produk di sebuah toko berdasarkan data historis yang berdistribusi eksponensial. Kemudian dibangkitkan bilangan random untuk menunjukkan jumlah pelanggan yang dibangkitkan sesuai posisi interval distribusinya



Model Simulasi Dinamik

- Model simulasi dinamik merepresentasikan sistem dari waktu ke waktu,
 - contoh, simulasi sebuah bank dalam rentang jam kerja tertentu. Namun harus diperhatikan bahwa model simulasi dinamis dalam pengertian ini berbeda dengan model simulasi sistem dinamis (dynamic system)



Model Simulasi Deterministik & Stokastik

- Model simulasi deterministik adalah model simulasi yang tidak memiliki variable random dalam inputnya.
 - Contoh, simulasi kedatangan pasien seorang dokter praktek yang telah diatur jadwal pelayanannya.
- Model simulasi stokastik adalah model simulasi yang memiliki satu atau beberapa variabel random dalam inputnya. Random input ini akan menghasilkan output yang random pula.
 - Contoh, Simulasi layanan teller bank



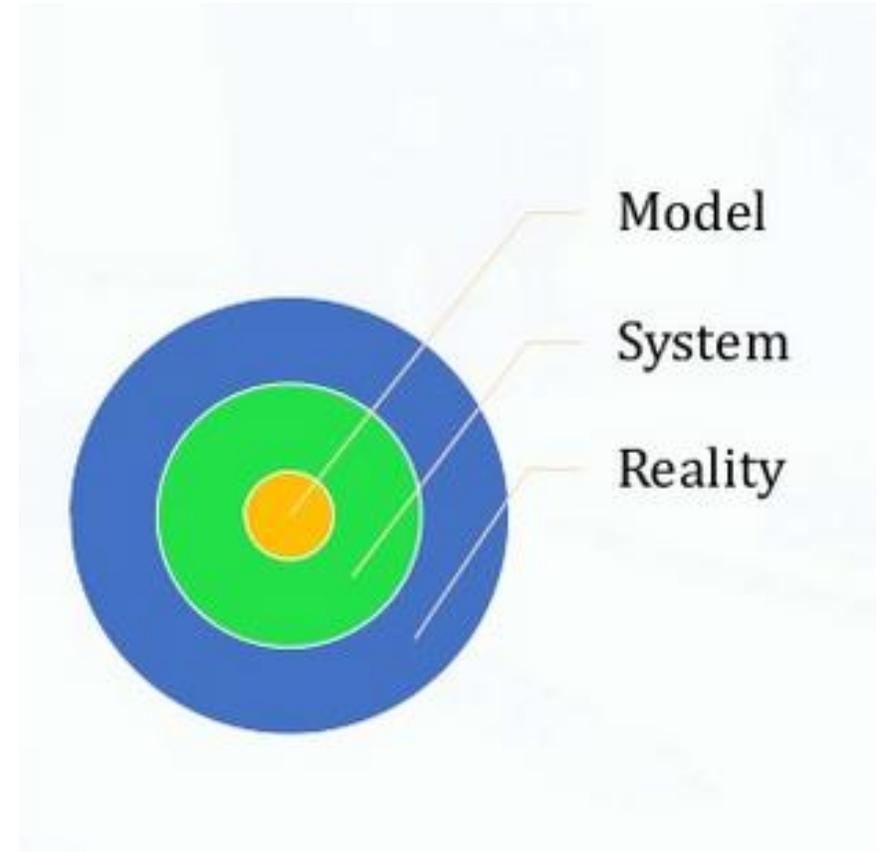
Model Simulasi Diskrit & Kontinyu

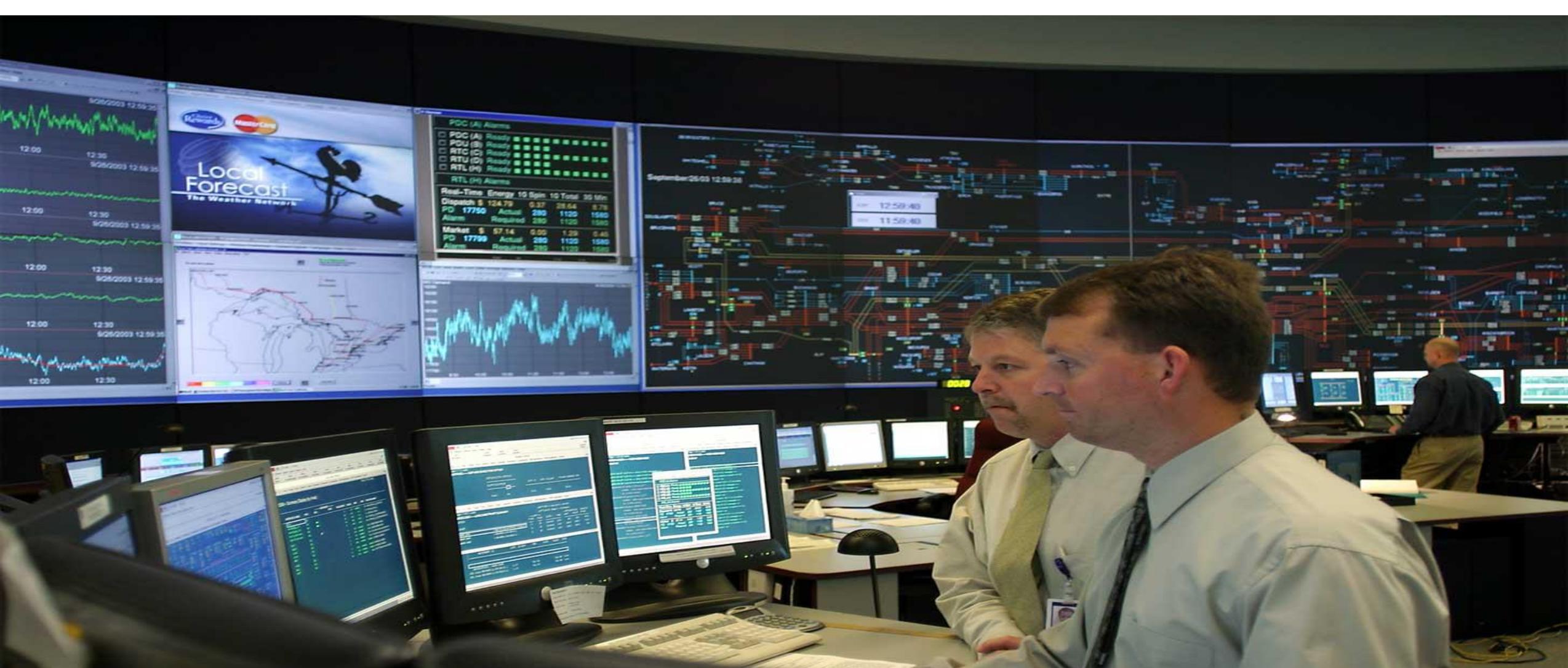
- Model simulasi diskrit adalah model simulasi yang status variabelnya berubah secara diskrit pada satu waktu tertentu.
 - Contohnya, simulasi layanan teller bank, dimana jumlah pelanggan yang menunggu/antri berubah secara diskrit dari waktu ke waktu.
- Model simulasi kontinu adalah model simulasi yang status variabel berubah secara kontinu dari waktu ke waktu.
 - Contoh Simulasi permukaan air bendungan



Hubungan Realitas, Sistem dan Model

- System is an Identification based on Purpose of study of reality
- Model is an abstraction based on efficiency of representation
- Simulation is part of modelling, for convincing or further study of system modelling





3. JENIS JENIS SIMULASI

Model Simulasi Statik

- Model simulasi statik dikenal juga dengan nama Simulasi Monte Carlo yang merepresentasikan sebuah sistem pada suatu waktu tertentu.
 - contoh, simulasi jumlah pelanggan yang membeli suatu produk di sebuah toko berdasarkan data historis yang berdistribusi eksponensial. Kemudian dibangkitkan bilangan random untuk menunjukkan jumlah pelanggan yang dibangkitkan sesuai posisi interval distribusinya



Model Simulasi Dinamik

- Model simulasi dinamik merepresentasikan sistem dari waktu ke waktu,
 - contoh, simulasi sebuah bank dalam rentang jam kerja tertentu. Namun harus diperhatikan bahwa model simulasi dinamis dalam pengertian ini berbeda dengan model simulasi sistem dinamis (dynamic system)



Model Simulasi Deterministik & Stokastik

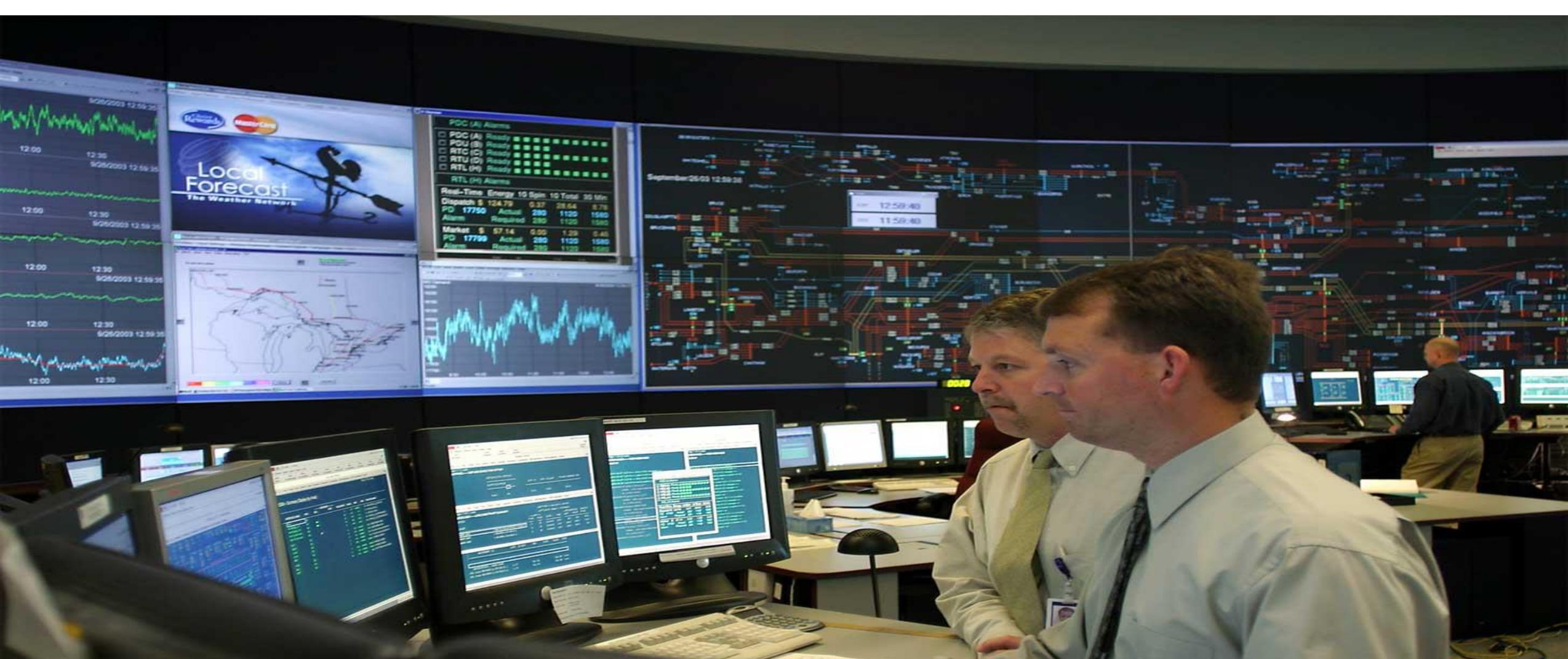
- Model simulasi deterministik adalah model simulasi yang tidak memiliki variable random dalam inputnya.
 - Contoh, simulasi kedatangan pasien seorang dokter praktek yang telah diatur jadwal pelayanannya.
- Model simulasi stokastik adalah model simulasi yang memiliki satu atau beberapa variabel random dalam inputnya. Random input ini akan menghasilkan output yang random pula.
 - Contoh, Simulasi layanan teller bank



Model Simulasi Diskrit & Kontinyu

- Model simulasi diskrit adalah model simulasi yang status variabelnya berubah secara diskrit pada satu waktu tertentu.
 - Contohnya, simulasi layanan teller bank, dimana jumlah pelanggan yang menunggu/antri berubah secara diskrit dari waktu ke waktu.
- Model simulasi kontinu adalah model simulasi yang status variabel berubah secara kontinu dari waktu ke waktu.
 - Contoh Simulasi permukaan air bendungan





4. SIMULASI KEJADIAN DISKRIT

Pengertian

“sebagai kejadian yang melampaui waktu yang representatif dimana state (keadaan) variabel berubah seketika dan terpisah per titik waktu”.

“Dalam istilah matematik disebut sebagai sistem yang dapat berubah hanya pada bilangan yang dapat dihitung per titik waktu”.



Contoh *DES*

1. Simulasi pada sistem antrian pelayanan tunggal (Simulation of a Single-server Queueing System), pada pelayanan kasir di pertokoan (supermarket), Teller pada pelayanan nasabah perbankan dan ruang informasi pada perkantoran atau hotel.”
2. Simulasi pada sistem inventori/pegudangan.



Kejadian yang biasa terdapat dalam *DES*:

- Kedatangan sebuah entitas ke sebuah stasiun kerja (workstation)
- Kegagalan Resource
- Selesainya sebuah aktivitas
- Akhir Sebuah Shift



Komponen dan Penyusun *DES*

1. Kondisi Sistem

Kumpulan kondisi variabel yang dibutuhkan untuk menjelaskan suatu sistem dalam waktu tertentu

2. Waktu Simulasi

Variabel yang menggunakan nilai saat ini dari waktu didalam simulasi

3. Event List

Daftar yang mengandung waktu berikutnya ketika masing-masing tipe event akan terjadi



Komponen dan Penyusun *DES*

4. Statistical Counter

Variabel yang digunakan untuk menyimpan informasi statistik mengenai performansi sistem

5. Inisialisasi Rutin

Bagian program untuk menginisialisasi model simulasi pada $t=0$

6. Time Routine

Bagian program yang menentukan event berikutnya dari daftar event (event list) dan kemudian mempercepat waktu simulasi ke waktu ketika event terjadi



Komponen dan Penyusun *DES*

7. Event Routine

Bagian program yang memperbarui kondisi sistem ketika tipe suatu event tertentu terjadi (hanya ada satu event routine untuk masing-masing event type)

8. Library Routine

Himpunan bagian program yang digunakan untuk menghasilkan pengamatan random dari distribusi peluang tertentu sebagai bagian dari model simulasi

9. Report Generator

Bagian program computer untuk mengestimasi pengukuran yang diinginkan dari performansi dan laporan produksi ketika simulasi selesai



Komponen dan Penyusun *DES*

10. Program Utama

Bagian program yang menentukan waktu rutin yang digunakan untuk menentukan *event* berikutnya kemudian tranfers kontrol ke *event* terkait rutin untuk memperbarui status sistem tepat. Program utama juga mengecek pemutusan dan menentukan laporan generator ketika simulasi telah selesai



Contoh simulasi sistem diskrit sederhana

Misalkan suatu BPR “X” hanya mengoperasikan 1 orang teller untuk melayani nasabah, dimana waktu kedatangan nasabah ke BPR “X” berkisar setiap 1 sampai 10 menit. Seorang nasabah dilayani sekitar 1 sampai 6 menit. Akan dilakukan simulasi terhadap 6 orang nasabah untuk menghitung kinerja pelayanan seperti persentase waktu kosong teller, rata-rata waktu yang diperlukan nasabah untuk dilayani dan sebagainya. Misalkan waktu kedatangan dan waktu pelayanan dari 6 nasabah tersebut adalah sebagai berikut :



Contoh :

Nasabah Ke	Waktu Antar Kedatangan (Menit)
1	-
2	8
3	6
4	1
5	8
6	3



Contoh :

Nasabah Ke	Waktu Pelayanan (Menit)
1	4
2	1
3	4
4	3
5	2
6	4



Simulasi Kedatangan

Asumsikan bahwa nasabah 1 datang pada menit ke 0 dan 8 menit kemudian datang nasabah ke 2. Jadi nasabah ke 2 datang pada menit ke 8. Sedang nasabah ke 3 datang 6 menit setelah nasabah ke 2, jadi nasabah ke 3 datang pada menit ke 14. Dengan analogi yang sama, nasabah ke 4 datang pada menit ke 15, nasabah ke 5 datang pada menit ke 23, dan seterusnya.



Simulasi Kedatangan

Nasabah Ke	Waktu Antar Kedatangan (Menit)	Waktu Kedatangan (Menit Ke)
1	-	0
2	8	8
3	6	14
4	1	15
5	8	23
6	3	26



Simulasi Pelayanan

- Nasabah 1 datang pada saat ke 0 dan langsung dilayani oleh teller selama 4 menit. Karena datang pertama kali, maka nasabah 1 tidak memerlukan waktu tunggu. Waktu pelayanan nasabah 1 dimulai menit ke 0 dan berakhir menit ke 4. Nasabah 1 berada dalam system selama 4 menit.
- Nasabah 2 datang pada menit ke 8, berarti teller mempunyai delay (idle time) selama 4 menit. Nasabah ke 2 dilayani mulai menit ke 8 dan karena waktu layanan nasabah 2 adalah 1 menit maka layanan berakhir pada menit ke 9. Nasabah 2 tidak memerlukan waktu tunggu untuk dilayani, karena teller sebelumnya dalam keadaan idle. Nasabah 2 berada dalam system selama 1 menit.
- Nasabah 3 datang pada menit ke 14, berarti teller mempunyai idle time selama 5 menit. Nasabah 3 dilayani mulai menit ke 14 dan berakhir pada menit ke 18. Nasabah 3 tidak memerlukan waktu tunggu untuk dilayani dan berada dalam system selama 4 menit.



Simulasi Pelayanan

Nasa bah Ke	Waktu Antar Kedatangan	Waktu Kedata ngan	Waktu Pelaya nan	Waktu Mulai Dilayani	Waktu Selesai Dilayani	Waktu Menunggu (Antrian)	Waktu Dalam Sistem	Idle Time Teller
1	-	0	4	0	4	0	4	0
2	8	8	1	8	9	0	1	4
3	6	14	4	14	18	0	4	5
4	1	15	3	18	21	3	6	0
5	8	23	2	23	25	0	2	2
6	3	26	4	26	30	0	4	1

- Jumlah waktu pelayanan = 18
- Jumlah waktu tunggu (Antrian) = 3
- Jumlah waktu dalam sistem = 21
- Jumlah waktu idle teller = 12
- Waktu selesai dilayani terakhir = 30



Simulasi Pelayanan

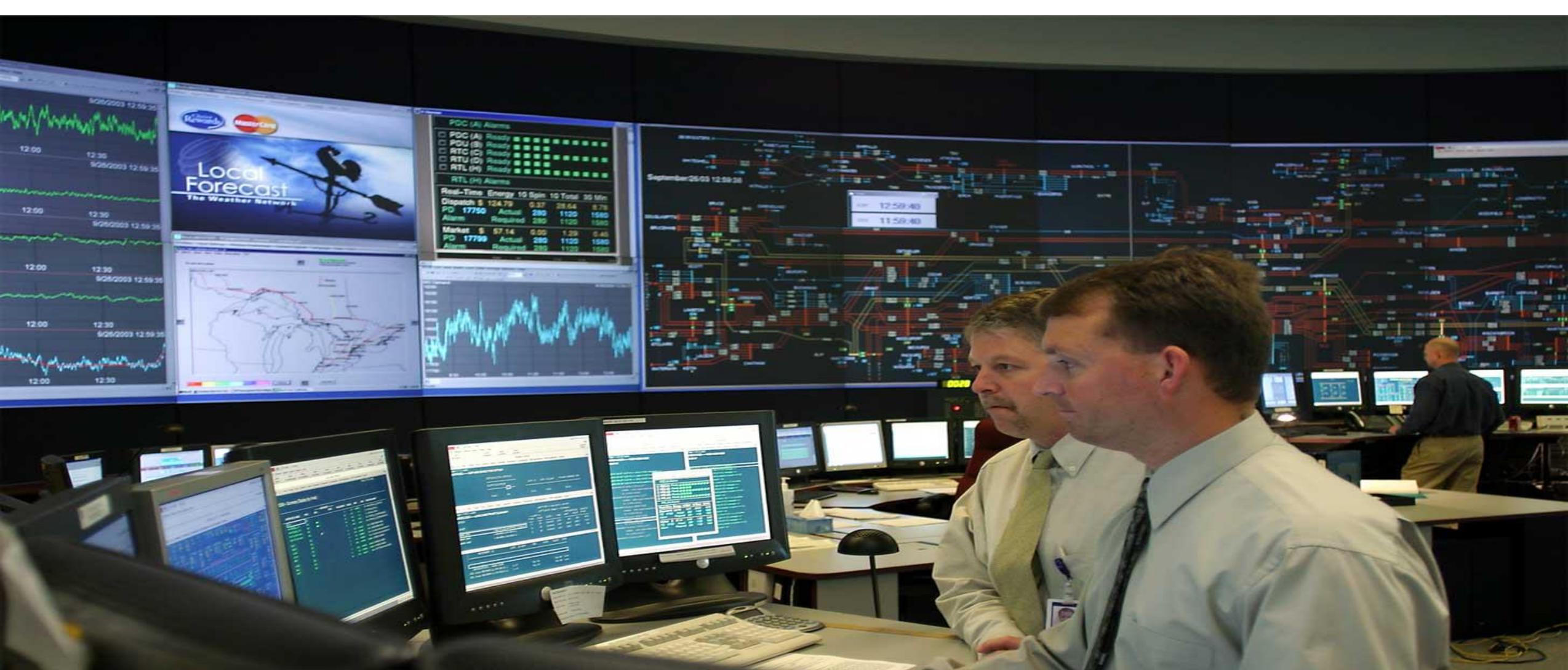
- Rata2 waktu tunggu = $3/6 = 0.5$ menit
- Peluang nasabah harus menunggu untuk dilayani : $6(\text{maksimum waktu dalam sistem})/6(\text{jumlah nasabah}) = 1$ (100%)
- Rata-rata waktu yang dibutuhkan nasabah dalam system : $21/6 = 3.5$ menit
(Rata-rata waktu menunggu + rata-rata waktu pelayanan)
- Rata-rata waktu pelayanan : $18/6 = 3$ menit
- Persentase teller menganggur : $(3/30) * 100\% = 10\%$
- Rata-rata waktu antar kedatangan nasabah : $26/5 = 5.2$ menit
- Rata-rata waktu tunggu nasabah yang mengantri : $3/6 = 0.5$ menit



Hasil

Berdasarkan hasil simulasi tersebut tampak bahwa sistem pelayanan nasabah pada BPR “X” belum bagus, sebab jumlah nasabah yang harus mengantri untuk mendapatkan pelayanan terhitung sangat tinggi yaitu 100% walaupun tingkat menganggur bagi teller hanya 10%.





5. PERANCANGAN AWAL MODEL SIMULASI

Langkah-langkah Melakukan Studi Simulasi

1

Formulasi Masalah Dan Perencanaan Studi

Studi diawali dengan pernyataan jelas tentang pokok masalah dan tujuan penelitian yang ingin dicapai. Setelah itu mempertimbangkan keterbatasan sarana dan prasarana yang tersedia



Langkah-langkah Melakukan Studi Simulasi

1

Formulasi Masalah Dan Perencanaan Studi

Menentukan :

- ✓ Model yang digunakan
- ✓ Kriteria performansi yang akan dipakai
- ✓ Kerangka konfigurasi sistem yang akan ditinjau



Langkah-langkah Melakukan Studi Simulasi

2

Pengumpulan Data Dan Perancangan Model

Pengumpulan data dan informasi dari sistem yang ditinjau diperlukan untuk mengetahui bagaimana sistem bekerja dan menentukan distribusi peluang bagi proses random yang digunakan dalam model. Kekurangan data akan mengurangi keakuratan model dan sebaliknya data yang terlalu kecil akan membutuhkan biaya besar dan waktu pengumpulan yang lama.



Langkah-langkah Melakukan Studi Simulasi

3

Validasi Model

Tahap ini dilakukan dengan melakukan pengecekan asumsi-asumsi yang ditetapkan dalam pembuatan model serta melibatkan ahli yang mengenal sistem dengan baik



Langkah-langkah Melakukan Studi Simulasi

4

Penyusunan Program Komputer Dan Verifikasi

Pemilihan perangkat lunak yang akan digunakan dalam simulasi mempunyai pengaruh yang besar terhadap kesuksesan penelitian, yaitu dalam hal keakuratan model, validitas model dan waktu eksekusi, dan waktu penyelesaian penelitian secara keseluruhan.



Langkah-langkah Melakukan Studi Simulasi

4

Penyusunan Program Komputer Dan Verifikasi

Beberapa teknik untuk melakukan verifikasi program antara lain :

- Melakukan pelacakan jalannya program (*trace*)
- Pengembangan program dalam bentuk sub program atau modular
- Menggunakan *interactive debugger*
- Meneliti kelayakan hasil program
- Menggunakan estimasi



Langkah-langkah Melakukan Studi Simulasi

5

Uji Coba Program

Uji coba program dilakukan untuk keperluan validasi pada tahap berikutnya.



Langkah-langkah Melakukan Studi Simulasi

6

Validasi Program

Hasil uji coba program diteliti kembali untuk mengetahui apakah ada kesalahan dalam program atau model yang digunakan



Langkah-langkah Melakukan Studi Simulasi

6

Validasi Program

cara yang dapat dilakukan antara lain :

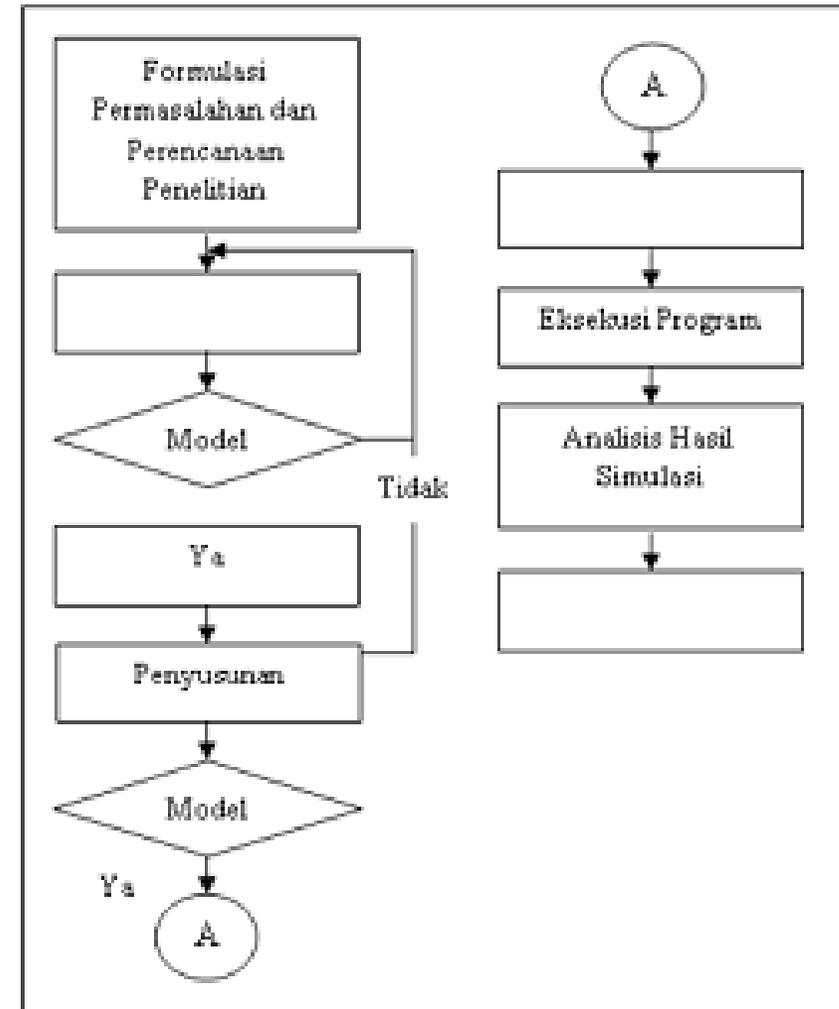
- ❖ Menguji sensitivitas output model terhadap perubahan input.
- ❖ Membandingkan output simulasi dengan performansi sistem di masa lalu (data historis). Jika hasil simulasi dengan data historis tidak berbeda secara signifikan, maka dapat dikatakan model simulasi sudah valid.



Langkah-langkah Melakukan Studi Simulasi

6

Validasi Program



Langkah-langkah Melakukan Studi Simulasi

7

Perancangan Eksperimen

Pada tahap ini diputuskan perancangan sistem seperti apa yang akan disimulasikan dari beberapa alternatif yang mungkin ada.

Perlu

Ditentukan

- ✓ Panjang replikasi dalam setiap eksekusi program
- ✓ Jumlah replikasi
- ✓ Inisiasi program
- ✓ Panjang periode *transient*



Langkah-langkah Melakukan Studi Simulasi

8

Eksekusi Program

Eksekusi program dilakukan sesuai dengan perancangan eksperimen yang telah dibuat.



Langkah-langkah Melakukan Studi Simulasi

9

Analisis Output Simulasi

Data output simulasi digunakan untuk mengestimasi kriteria performansi sistem yang diteliti. Hasil estimasi ini kemudian digunakan untuk menjawab tujuan studi.



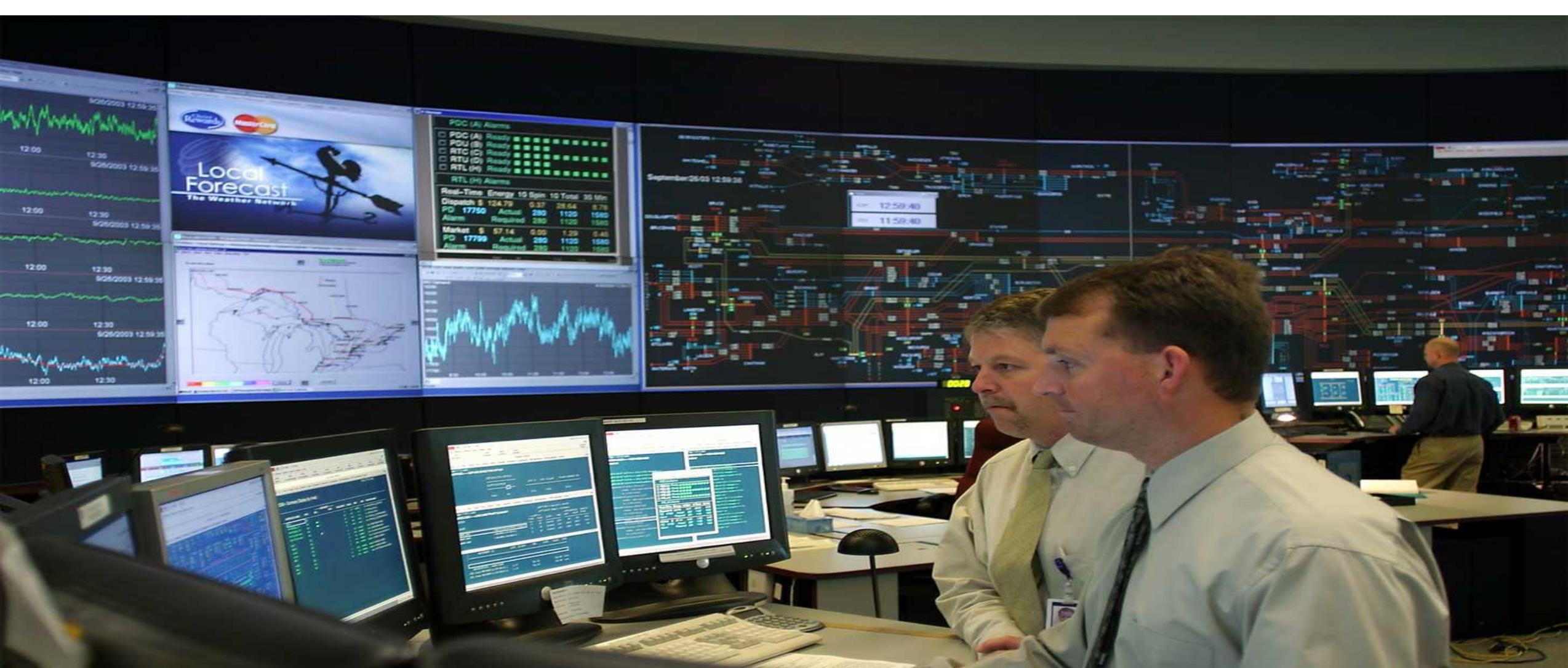
Langkah-langkah Melakukan Studi Simulasi

10

Dokumentasi, Presentasi Dan Implementasi

Dokumentasi yang baik diperlukan karena tidak jarang model simulasi yang telah dibuat akan dipakai untuk lebih dari satu aplikasi. Akhirnya hasil dari studi simulasi perlu diimplementasikan, untuk itu kredibilitas model simulasi yang dibangun harus tinggi agar dapat digunakan secara nyata.





6. STRUKTUR DASAR PROGRAM SIMULASI



Sejarah *Software* Komputer

Bentuk paling primitif dari perangkat lunak, menggunakan aljabar *Boolean*, yang di representasikan sebagai *binary digit (bit)*, yaitu *1 (benar / on)* atau *0 (salah / off)*, cara ini sudah pasti sangat menyulitkan, sehingga orang mulai mengelompokkan bit tersebut menjadi *nibble (4 bit)*, *byte (8 bit)*, *word (2 byte)*, *double word (32 bit)*.



Sejarah *Software* Komputer

“Kelompok-kelompok bit ini disusun ke dalam struktur instruksi seperti penyimpanan, transfer, operasi aritmatika, operasi logika, dan bentuk bit ini diubah menjadi kode-kode yang dikenal sebagai assembler

Kode-kode mesin sendiri masih cukup menyulitkan karena tuntutan untuk dapat menghafal kode tersebut dan format (aturan) penulisannya yang cukup membingungkan, dari masalah ini kemudian lahir bahasa pemrograman tingkat tinggi yang seperti bahasa manusia (bahasa Inggris).”



Bahasa Pemrograman



Bahasa pemrograman tingkat tinggi ini memiliki beberapa jenis yaitu :

- Bahasa C,
- Bahasa C++, dan
- Pascal



Bahasa Pemrograman

Bahasa C

Bahasa C adalah salah satu dari banyak bahasa pemrograman desktop yang cukup populer dan mempunyai cukup banyak keunggulan.

Bahasa C adalah sebuah bahasa yang sifatnya kompleks dan membangun logika atau algoritma. Software Development Kit untuk Windows juga dirancang / tulis dalam bahasa C.

Bahasa C berifat portable, yaitu dengan sedikit atau tanpa perubahan, suatu program yang dirancang / tulis dengan bahasa C dapat dijalankan pada komputer lain.



Bahasa Pemrograman

Bahasa C++

“Bahasa Pemrograman C++ ini dibuat oleh Bjarne Stroustrup pada tahun 1980 yang usut punya usut ternyata ini adalah Bahasa pengembangan Bahasa C yang lahir lebih dahulu pada tahun 1970-an yang dibuat oleh Dennis Ritchie di Bell Labs.”



Bahasa Pemrograman

Bahasa C++

Ternyata C++ memiliki sifat Pemrograman berorientasi objek karena, untuk menyelesaikan masalah, C++ melakukan langkah pertama dengan menjelaskan class-class yang merupakan anak class yang dibuat sebelumnya sebagai abstraksi dari object-object fisik, Class tersebut berisi keadaan object, anggotaanggotanya dan kemampuan dari objectnya, Setelah beberapa Class dibuat kemudian masalah dipecahkan dengan Class.



Bahasa Pemrograman

Pascal

Pascal Pertama kali di buat oleh Profesor Niklaus Wirth, seorang anggota International Federation of Information Processing (IFIP) pada tahun 1971.

Pascal merupakan bahasa yang terstruktur tersusun atas sejumlah blok. Blok-blok yang kecil selanjutnya dapat dapat dipakai untuk membuat blok yang lebih besar. Blok ini lebih dikenal dengan sebutan subprogram, yang dibedakan atas prosedur dan fungsi.



Software Simulasi

“Proses simulasi secara spesifik baik untuk aktivitas permesinan, antrian dan lainnya dapat dilakukan dengan bantuan **Promodel, stella, i-think, Arena, Dynamo, Vensim** (*Ventana Simulation*), **Powersim, simul8.com, awesim, GPSS, SIMAN, SIMSCRIPT II.5, SLAM II** hingga modular **Simquick** yang terdapat dalam *microsoft excel*. **Bahasa C/C++, pascal, fortran, Basic** juga dapat digunakan untuk membuat alat bantu simulasi.”



Software Simulasi

Banks (2010) mengatakan ada banyak fitur yang relevan saat memilih software simulasi. Beberapa hal yang harus dievaluasi saat memilih perangkat lunak simulasi, meliputi:

- ✓ Keakuratan dan tingkat detail, kemudahan untuk dipelajari dan penerapannya pada aplikasi yang akan digunakan.
- ✓ Kecepatan saat proses eksekusi.
- ✓ Waspada terhadap klaim dan demonstrasi iklan saat penggunaan.
- ✓ Vendor mampu memberikan versi sederhana dari masalah.



Software Simulasi

Saat memilih software simulasi, fitur yang harus dimiliki software harus mampu mendeskripsikan,

- fitur model,
- runtime environment,
- fitur layout animasi,
- fitur output,
- dukungan vendor, dan
- dokumentasi produk yang terdapat pada software simulasi tersebut



Software Simulasi

PROMODEL

“ProModel (Production Modeler) yang dikeluarkan oleh ProModel Corporation sebagai perangkat simulasi untuk berbagai macam jenis model sistem manufaktur dan jasa. Penggunaan pada sistem jasa seperti pada rumah sakit, call centers, warehouse operational, transportation systems, queuing system, information systems, customer service management, dan proses bisnis lainnya dapat dimodel dengan efisien dan cepat.”



Software Simulasi

PROMODEL

“Software ProModel yang digunakan adalah software berbasis Windows dengan hasil dari simulasi berupa data-data statistik yang dapat digunakan untuk menganalisa model dari sistem yang ada. Software ini dikembangkan secara spesifik untuk manajer operasional dalam mengembangkan desain proses operasional yang ada.”



Software Simulasi

PROMODEL

“Menurut Harrel (2011), ProModel merupakan software simulasi yang dirancang untuk memodelkan sistem dengan proses discrete-event dan menyediakan opsi pelaporan secara periodik dan statistik yang diolah berdasarkan periode yang dapat memberikan gambaran aktivitas sistem yang lebih lengkap pada saat keseluruhan simulasi”.



Keuntungan menggunakan software ProModel :

1. Mampu mendeteksi bottleneck yang terjadi pada proses produksi dan dapat langsung dieliminasi.
2. Mampu mengembangkan suatu proses produksi yang efisien.
3. Mengurangi lead time yang diperlukan.
4. Meningkatkan utilisasi sumber daya.
5. Mengurangi terjadinya penumpukan inventory.

Software Simulasi

VENSIM

“Vensim adalah sebuah perangkat lunak simulasi yang dibuat oleh Ventana Systems, Inc (Harvard , Massachusetts). Tujuannya adalah untuk membantu perusahaan untuk menemukan solusi optimal untuk berbagai situasi yang membutuhkan analisis dan di mana itu diperlukan untuk mengetahui semua hasil yang mungkin dari implementasi masa depan atau keputusan.”



Software Simulasi

Vensim mampu mensimulasikan perilaku dinamis dari sistem, yang tidak mungkin untuk menganalisis tanpa yang sesuai perangkat lunak simulasi, karena mereka tidak bisa ditebak karena banyak pengaruh, umpan balik dll Ini membantu dengan kausalitas loop identifikasi dan menemukan poin memanfaatkan.

The logo for VENSIM is a blue hexagonal shape with the word "VENSIM" written in white, bold, uppercase letters in the center.

VENSIM

Software Simulasi



ARENA

“Arena adalah suatu perangkat lunak simulasi dan otomasi yang dikembangkan oleh System Modelling. Dalam Arena, pengguna membangun model eksperimen dengan menggunakan modul-modul yang menyatakan proses atau logika. Garis penghubung digunakan untuk menyatakan hubungan antar modul dan atau menyatakan aliran entitas.



Software Simulasi

Arena dapat terintegrasi dengan baik dengan teknologi Microsoft termasuk Visual Basic. Arena juga mendukung diagram alir yang dibuat dengan menggunakan Microsoft Visio, membaca Excel dan Acces.

Di dalam Arena akan disimulasikan model yang telah dibentuk sebelumnya dengan input data primer maupun sekunder sebagai resources dalam pengoperasiannya.



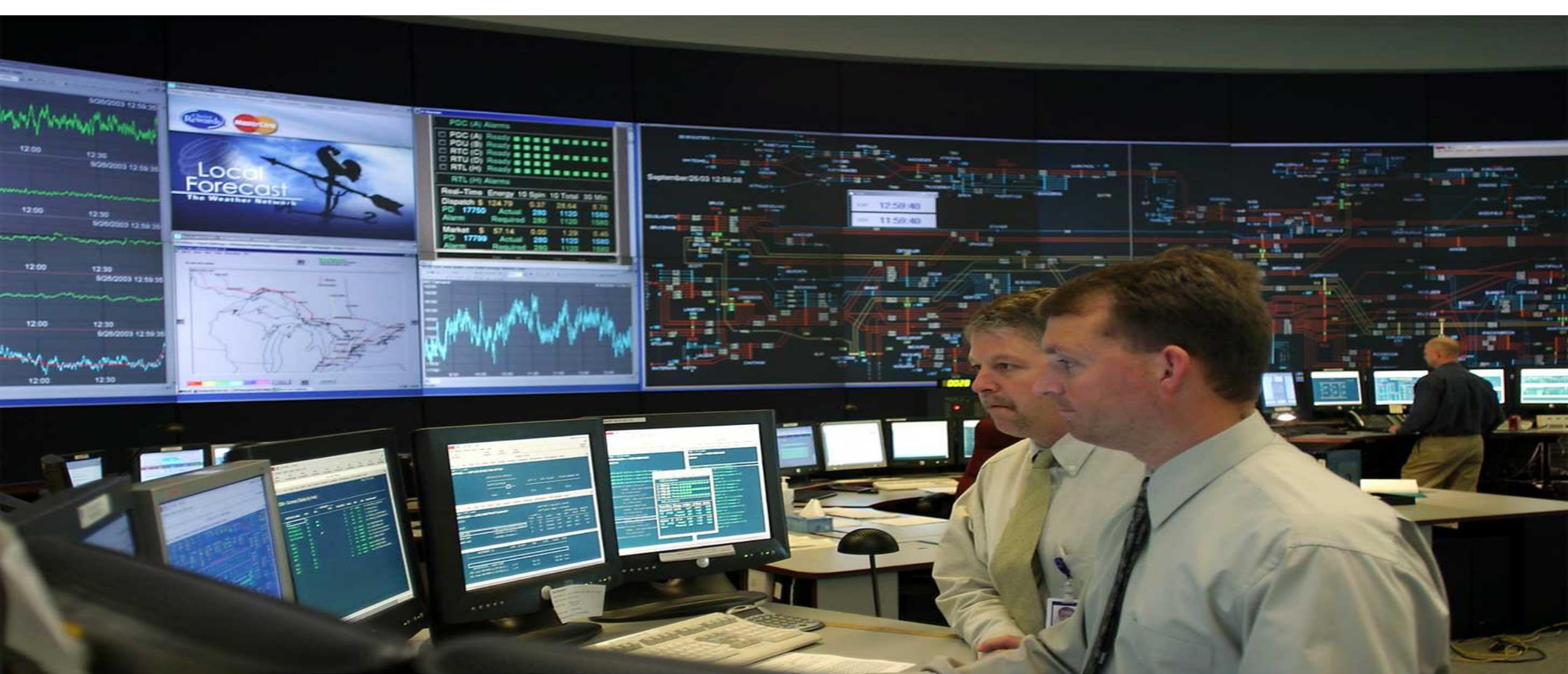
Software Simulasi

ARENA

Keuntungan menggunakan *software arena* adalah :

1. Menganalisa keseluruhan *item yang diinputkan dari level awal sampai level* akhir.
2. Dapat digunakan untuk menganalisis bisnis seperti: industri global, perbankan, asuransi keuangan, dan lain-lain.
3. Penggambaran aliran proses nyata untuk mempermudah proses rekonstruksi proses yang lama dengan perencanaan yang baru.





7. PEMBANGKIT BILANGAN RANDOM



PEMBANGKITAN BILANGAN RANDOM (Random Number Generator)

1

Cara Memperoleh

a. Zaman Dahulu, dengan cara:

- Melempar dadu
- Mengocok kartu

b. Zaman Modern (>1940), dengan cara:

Membentuk bilangan acak secara numerik/aritmatik (menggunakan komputer), disebut “Pseudo Random Number” (bilangan pseudo acak).



Pembangkit Bilangan Random

2

Pembangkitan Bilangan Acak Harus

- ✓ Berdistribusi uniform(0,1) dan tidak berkorelasi antar bilangan
- ✓ Membangkitkan cepat, storage tidak besar
- ✓ Dapat di “reproduce”
- ✓ Periode besar, karena mungkin bilangan acak dibangkitkan berulang.



Pembangkit Bilangan Random

3

Pengertian

- Bilangan acak adalah bilangan yang tidak dapat diprediksi kemunculannya.
- Tidak ada komputasi yang benar-benar menghasilkan deret bilangan acak secara sempurna.
- Bilangan acak yang dibangkitkan oleh komputer adalah bilangan acak semu (Pseudo Random Number), karena menggunakan rumus-rumus matematika.
- Banyak algoritma atau metode yang dapat digunakan untuk membangkitkan bilangan acak.
- Bilangan acak dapat dibangkitkan dengan pola tertentu yang dinamakan dengan distribusi mengikuti fungsi distribusi yang ditentukan.



Pembangkit Bilangan Random

4

Sifat Sifat

Independen : tiap variabelnya harus bebas dari ketentuan, seperti:

Z_{i-1} = merupakan hasil akhir

Z_0 = merupakan angka pertama yang bebas tertentu

a = merupakan angka pertama yang bebas dengan ketentuan tersendiri

c = merupakan angka bebastetapi tidak ada hubungan tertentu dengan m



Pembangkit Bilangan Random

4

Sifat Sifat

Uniform : suatu distribusi yang umum (distribusi probabilitas) dan sama untuk semua besaran yang dikeluarkan/diambil. Hal ini berarti bahwa diusahakan probabilitasnya sama untuk setiap penarikan random number tersebut.



Pembangkit Bilangan Random

4

Sifat Sifat

Dense: Density Probabilitas Distribution harus mengikuti syarat probabilitas (antara 0 dan 1). Hal ini berarti dalam penarikan angka-angka yang dibutuhkan dari Random Number Generator cukup untuk banyak dan dibuat sedemikian rupa sehingga $0 \leq R.N. \leq 1$.



Pembangkit Bilangan Random

4

Sifat Sifat

Efficient : artinya dapat cukup Sederhana dan dalam menggunakan cara ini harus terlebih dahulu memilih angka-angka untuk variabel-variabel yang cocok. Hal ini berarti dalam penarikan random number tersebut harus dapat menentukan angka-angka untuk variabel yang sesuai sehingga dapat berjalan terus-menerus.



Pembangkit Bilangan Random

5

Penentuan Random Number

- ❖ Tabel Random Number, tabel ini sudah banyak ditemukan mulai dari enam digit sampai dengan belasan digit.
- ❖ Electronic Random Number, number ini banyak juga dipergunakan dalam percobaan penelitian.
- ❖ Conguential Pseudo Random Number Generator, yang terdiri dari tiga bagian:
 - ✓ Linear Congruential Generator (LCG)
 - ✓ Multiplicative Random Number Generator
 - ✓ Mixed Congruential Random Number Generator



Pembangkit Bilangan Random

5

Linear Congruential Generator (LCG)

a. Metode ini digunakan untuk membangkitkan bilangan acak dengan distribusi uniform.

b. Pseudo RNG, berbentuk:

$$Z_i = (aZ_{i-1} + c) \bmod m$$

Dimana :

Z_i = bilangan acak ke-i dari deretnya

Z_{i-1} = bilangan acak sebelumnya

a = faktor pengali

c = increment

m = modulus

kunci pembangkit adalah Z_0 yang disebut umpan (seed)



Contoh 1. LCG:

Membangkitkan bilangan acak sebanyak 8 kali dengan $a = 2$, $c = 7$, $m = 10$, dan $Z_0 = 2$

$$Z_1 = (2 \cdot 2 + 7) \bmod 10 = 1$$

$$Z_2 = (2 \cdot 1 + 7) \bmod 10 = 9$$

$$Z_3 = (2 \cdot 9 + 7) \bmod 10 = 5$$

$$Z_4 = (2 \cdot 5 + 7) \bmod 10 = 7$$

$$Z_5 = (2 \cdot 7 + 7) \bmod 10 = 1$$

$$Z_6 = (2 \cdot 1 + 7) \bmod 10 = 9$$

$$Z_7 = (2 \cdot 9 + 7) \bmod 10 = 5$$

$$Z_8 = (2 \cdot 5 + 7) \bmod 10 = 7$$

Bilangan acak yang dibangkitkan adalah :

1 9 5 7 1 9 5 7

jadi terjadi pengulangan bilangan secara periodik (4).



Contoh 2. LCG:

Membangkitkan bilangan acak sebanyak 8 kali dengan $a = 4$, $c = 7$, $m = 15$, dan $Z_0 = 3$

$$Z_1 = (4 \cdot 3 + 7) \bmod 15 = 4$$

$$Z_2 = (4 \cdot 4 + 7) \bmod 15 = 8$$

$$Z_3 = (4 \cdot 8 + 7) \bmod 15 = 9$$

$$Z_4 = (4 \cdot 9 + 7) \bmod 15 = 13$$

$$Z_5 = (4 \cdot 13 + 7) \bmod 15 = 14$$

$$Z_6 = (4 \cdot 14 + 7) \bmod 15 = 3$$

$$Z_7 = (4 \cdot 3 + 7) \bmod 15 = 4$$

$$Z_8 = (4 \cdot 4 + 7) \bmod 15 = 8$$

Bilangan acak yang dibangkitkan adalah :

4 8 9 13 14 4 8

jadi tidak terjadi pengulangan secara periodik.



Pembangkit Bilangan Random

5

Linear Congruential Generator (LCG)

- c. Terjadi pengulangan pada periode tertentu atau setelah sekian kali pembangkitan, hal ini adalah salah satu sifat pembangkitan dari metode ini dan PRNG pada umumnya.
- d. LCG mempunyai periode tidak lebih besar dari m , dan pada kebanyakan kasus periodenya kurang dari itu.



Pembangkit Bilangan Random

5

Linear Congruential Generator (LCG)

e. LCG mempunyai periode penuh ($m - 1$) jika memenuhi syarat berikut:

1. c relatif prima terhadap m
2. $a-1$ dapat dibagi dengan semua faktor prima dari m
3. $a-1$ adalah kelipatan 4 jika m adalah kelipatan 4
4. $m > \max(a, c, Z_0)$
5. $a > 0, c > 0$

f. Penentuan konstanta LCG (a, c dan m) sangat menentukan baik tidaknya bilangan acak yang diperoleh dalam arti memperoleh bilangan acak yang seakan-akan terjadi pengulangan.



Pembangkit Bilangan Random

5

Multiplicative Random Number Generator

$$Z_i = (a \cdot Z_{i-1}) \bmod m$$

Dimana:

- a. Bilangan pseudo dimulai dengan nilai awal Z_0 yang disebut benih.
- b. a & m : bilangan bulat positif tertentu
- c. $a \cdot Z_{i-1}$ dibagi dengan m dan sisanya diambil sebagai nilai Z_n
- d. Agar Z_n berperilaku acak yang dapat dipertanggungjawabkan maka:
 - Modulo m dipilih sebesar mungkin untuk memperbesar periode
 - a dipilih agar korelasi antar Z_n minimum
 - benih Z_0 : bilangan Bulat positif ganjil, $Z_0 < m$
 - Bilangan acak : $U = Z_n/m$



Pembangkit Bilangan Random

5

Multiplicative Random Number Generator

Untuk pemilihan nilai-nilai yang terbaik dijabarkan sebagai berikut:

- ✓ Pemilihan nilai: m (modulo) merupakan suatu angka integer yang cukup besar dan merupakan satu kata dari yang dipakai pada komputer.
- ✓ Pemilihan konstanta multipliter: a harus tepat.
- ✓ Pemilihan untuk Z_0 yang dikenal dengan : SEED = Z_0 mengharuskan relative belakangan prima terhadap m . Hal ini dapat diperhatikan dengan mudah apabila dicari untuk m adalah angka berpangkat 2. Dengan demikian untuk Z_0 adalah setiap angka- angka yang ganjil seperti : ISEED =12357 dapat diambil sembarang asalkan bilangan ganjil dan biasanya cukup besar.
- ✓ Bilangan c yang dipilih harus bukan merupakan kelipatan dari m dan juga harus bilangan ganjil.



Pembangkit Bilangan Random

5

Mixed Congruential Random Number Generator

- ✓ Pseudo Random Number ini dapat dirumuskan dengan:

$$Z_n = a_n Z_0 + a_{n-1} / a - 1 \cdot c + (\text{mod}.m)$$

- ✓ Rumus Pseudo Number generator ini adalah dengan syarat utama n harus sejumlah bilangan integer (bulat) dan lebih besar dari nol, rumus ini dikenal juga dengan nama 'Linier Congruential RNG'
- ✓ Namun apabila nilai $C = 0$ maka akan diperoleh rumus yang dikenal 'Multiplicative Congruen RNG'. Rumus multiplivative ini cukup baik untuk masa-masa yang akan datang karena sedikit sekali storage memori yang dibutuhkan.



Pembangkit Bilangan Random

5

Mixed Congruential Random Number Generator

d. Beberapa kondisi syarat-syaratnya sebagai berikut:

C = adalah bilangan relative prima terhadap n

$N = 1 \pmod{q}$ untuk setiap factor prima q dari m

$A = 1 \pmod{4}$ apabila 4 adalah suatu factor dari m

e. Kondisi 1 berarti bahwa pembagi umum yang terbesar dari c dan m adalah satu.

Dan kondisi ini mudah disari.

f. Kondisi 2 berarti:

$$a - q(a/q) = 1$$

Apabila $k = (a/q)$ akan dapat diperoleh untuk a , yaitu $a = 1 + qk$

Diman q adalah faktor prima dari m

g. Kondisi 3 : berarti $a = 1 + 4k$

Apabila k adalah integer. Artinya m bilangan bulat dapat dibagi 4



Pembangkit Bilangan Random

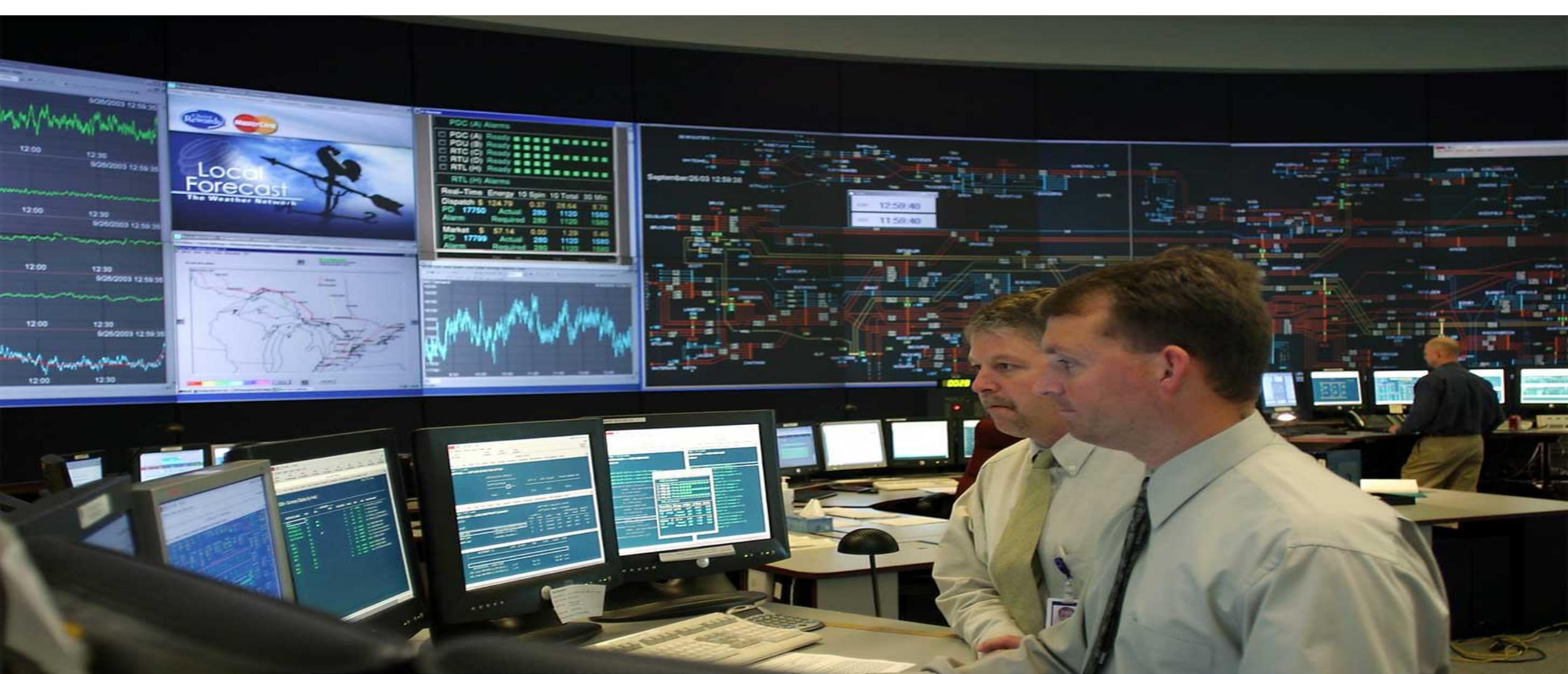
6

Penerapan

Simulasi kejadian "acak" ("random" events) dalam sebuah restorasi drive-through

- Waktu tiba mobil di jendela restoran drive-through
- Waktu yang diperlukan pengemudi untuk memesan
- Jumlah hamburger, minuman, dan kentang yang diorder
- Waktu yang diperlukan oleh restoran untuk menyiapkan pesanan
- Panjang rentetan bilangan acak dapat dibagi-bagi dalam segmen yang lebih kecil, yang disebut aliran/stream.
- Contoh : stream 1: pola kedatangan mobil ke jendela restoran drive-through; stream 2 : waktu yang diperlukan oleh pengemudi untuk memesan.





8. PEMBANGKIT VARIATES RANDOM

Random Variate

- Sistem stokastik seringkali memiliki waktu atau kuantitas nilai yang bervariasi dalam sebuah jangkauan dan berdasarkan densitas tertentu, sebagaimana didefinisikan oleh distribusi probabilitas
- Nilai yang dihasilkan dari sebuah distribusi disebut **random variate**



Membangkitkan *Random Variate*

Tipe metoda untuk membangkitkan bilangan acak dari distribusi yang diinginkan

- **Inverse transform**
- Komposisi
- Konvolusi



Membangkitkan *Random Variate*

Distribusi kontinu:

- Misalkan sebuah fungsi densitas probabilitas adalah $f(x)$
- Cari fungsi distribusi kumulatif dari X ; $F(x)$
- $U=F(x)$, dimana U adalah uniform $(0,1)$
- Hitung x dimana $x=F^{-1}(U)$



Membangkitkan *Random Variate*

Distribusi kontinu

Contoh: waktu antar kedatangan mobil ke jendela restoran *drive-through* terdistribusi eksponensial. Misalkan rata-rata waktu antar kedatangan mobil adalah β . Maka fungsi densitas probabilitas $f(x)$ dan fungsi distribusi kumulatif $F(x)$ adalah

$$f(x) = \begin{cases} \frac{1}{\beta} e^{-x/\beta} & \text{for } x > 0 \\ 0 & \text{elsewhere} \end{cases}$$

$$F(x) = \begin{cases} 1 - e^{-x/\beta} & \text{for } x > 0 \\ 0 & \text{elsewhere} \end{cases}$$



Membangkitkan *random variate*

Distribusi kontinu

Set $U=F(x)$ dan cari nilai x

$$U = 1 - e^{-x/\beta}$$

$$e^{-x/\beta} = 1 - U$$

$$\ln(e^{-x/\beta}) = \ln(1 - U)$$

$$-x/\beta = \ln(1 - U)$$

$$x = -\beta \ln(1 - U)$$



Contoh: Model simulasi dinamik, stokastik



(GAP PHOTO)



Contoh: Model simulasi dinamik, stokastik

- Pelanggan tiba menggunakan mesin ATM dengan waktu interval rata-rata 3,0 menit yang terdistribusi eksponensial
- Ketika pelanggan datang ke sistem, mereka bergabung dalam antrian untuk menunggu giliran menggunakan ATM. Antrian memiliki kapasitas tak terbatas untuk menampung pelanggan
- Pelanggan menghabiskan rata-rata 2,4 menit terdistribusi eksponensial pada mesin ATM untuk menyelesaikan transaksinya, yang disebut waktu jasa (*service time*) pada ATM



Contoh: Model simulasi dinamik, stokastik

- Simulasikan sistem kedatangan dan pemrosesan 25 pelanggan, dan estimasikan/perkirakan:
- Waktu tunggu yang diharapkan pelanggan di dalam antrian (waktu rata-rata pelanggan menunggu di dalam antrian untuk menggunakan ATM)
- Waktu yang diharapkan dalam sistem (waktu rata-rata pelanggan menunggu di dalam antrian ditambah dengan waktu rata-rata yang diperlukan untuk menyelesaikan transaksinya di ATM)



Contoh: Model simulasi dinamik, stokastik

- Sistem: ATM
- Entitas: pelanggan yang tiba di ATM untuk melakukan transaksi
- Resource: ATM yang melayani pelanggan dengan kapasitas untuk melayani satu pelanggan pada satu waktu
- Kontrol sistem yang mengatur bagaimana, kapan, dan dimana aktivitas dilakukan untuk ATM ini



Contoh: Model simulasi dinamik, stokastik

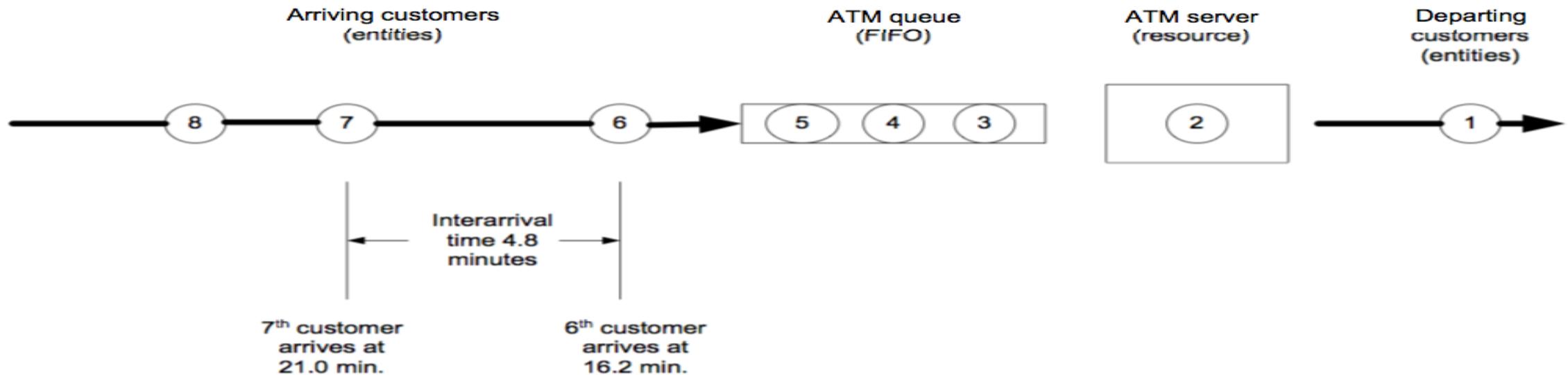


Figure: Descriptive drawing of the ATM system



TABLE 3.2 Spreadsheet Simulation of Automatic Teller Machine (ATM)

Arrivals to ATM			ATM Processing Time			ATM Simulation Logic						
Stream 1 (Z _{1i})	Random Number (U _{1i})	Interarrival Time (X _{1i})	Stream 2 (Z _{2i})	Random Number (U _{2i})	Service Time (X _{2i})	Customer Number (1)	Arrival Time (2)	Begin Service Time (3)	Service Time (4)	Departure Time (5) = (3) + (4)	Time in Queue (6) = (3) - (2)	Time in System (7) = (5) - (2)
0	3		122									
1	66	0.516	5	0.039	0.10	1	2.18	2.18	0.10	2.28	0.00	0.10
2	109	0.852	108	0.844	4.46	2	7.91	7.91	4.46	12.37	0.00	4.46
3	116	0.906	95	0.742	3.25	3	15.00	15.00	3.25	18.25	0.00	3.25
4	7	0.055	78	0.609	2.25	4	15.17	18.25	2.25	20.50	3.08	5.33
5	22	0.172	105	0.820	4.12	5	15.74	20.50	4.12	24.62	4.76	8.88
6	81	0.633	32	0.250	0.69	6	18.75	24.62	0.69	25.31	5.87	6.56
7	40	0.313	35	0.273	0.77	7	19.88	25.31	0.77	26.08	5.43	6.20
8	75	0.586	98	0.766	3.49	8	22.53	26.08	3.49	29.57	3.55	7.04
9	42	0.328	13	0.102	0.26	9	23.72	29.57	0.26	29.83	5.85	6.11
10	117	0.914	20	0.156	0.41	10	31.08	31.08	0.41	31.49	0.00	0.41
11	28	0.219	39	0.305	0.87	11	31.82	31.82	0.87	32.69	0.00	0.87
12	79	0.617	54	0.422	1.32	12	34.70	34.70	1.32	36.02	0.00	1.32
13	126	0.984	113	0.883	5.15	13	47.11	47.11	5.15	52.26	0.00	5.15
14	89	0.695	72	0.563	1.99	14	50.67	52.26	1.99	54.25	1.59	3.58
15	80	0.625	107	0.836	4.34	15	53.61	54.25	4.34	58.59	0.64	4.98
16	19	0.148	74	0.578	2.07	16	54.09	58.59	2.07	60.66	4.50	6.57
17	18	0.141	21	0.164	0.43	17	54.55	60.66	0.43	61.09	6.11	6.54
18	125	0.977	60	0.469	1.52	18	65.87	65.87	1.52	67.39	0.00	1.52
19	68	0.531	111	0.867	4.84	19	68.14	68.14	4.84	72.98	0.00	4.84
20	23	0.180	30	0.234	0.64	20	68.74	72.98	0.64	73.62	4.24	4.88
21	102	0.797	121	0.945	6.96	21	73.52	73.62	6.96	80.58	0.10	7.06
22	97	0.758	112	0.875	4.99	22	77.78	80.58	4.99	85.57	2.80	7.79
23	120	0.938	51	0.398	1.22	23	86.12	86.12	1.22	87.34	0.00	1.22
24	91	0.711	50	0.391	1.19	24	89.84	89.84	1.19	91.03	0.00	1.19
25	122	0.953	29	0.227	0.62	25	99.01	99.01	0.62	99.63	0.00	0.62
Average											1.94	4.26



Contoh: Model simulasi dinamik, stokastik

Replication	Z1 ₀	Z2 ₀
1	3	122
2	29	92

TABLE 3.3 Summary of ATM System Simulation Output

<i>Replication</i>	<i>Average Time in Queue</i>	<i>Average Time in System</i>
1	1.94 minutes	4.26 minutes
2	0.84 minutes	2.36 minutes
Average	1.39 minutes	3.31 minutes



Pembangkitan Random variate kontinu

- Uniform, $U(a,b)$
 - $X = F^{-1}(u) = a + (b - a) u$
 - Dibangkitkan $U \sim U(0,1)$
 - $X = a + (b-a) U$
- Exponensial
 - Dibangkitkan $U \sim U(0,1)$
 - $X = -\beta \ln U$



Pembangkitan random variate kontinu

- M-Erlang

- Digunakan konvolusi

- Dibangkitkan U_1, U_2, \dots, U_m sebagai IID $U(0,1)$
- Di kembalikan

$$X = \frac{-\beta}{m} \ln \left(\prod_{i=1}^m U_i \right)$$

- Gamma

- Dibangkitkan $b=(e+\alpha)/e$

1. Dibangkitkan $U_1 \sim U(0,1)$, $P = bU_1$, jika $P > 1$ ke step 3
2. $Y = P^{1/\alpha}$, dibangkitkan $U_2 \sim U(0,1)$, jika $U_2 \leq e^{-Y}$ maka $X = Y$, kalau tidak ke step 1
3. $Y = -\ln[(b-P)/\alpha]$, dibangkitkan $U_2 \sim U(0,1)$. Jika $U_2 \leq Y^{\alpha-1}$ maka $X = Y$, kalau tidak kembali ke step 1



Pembangkitan random variate kontinu

- Weibull
 - Dibangkitkan $U \sim U(0,1)$
 - $X = \beta(-\ln U)^{1/\alpha}$
- Normal
 - Dibangkitkan U_1 dan U_2 sebagai IID $U(0,1)$, $V_1 = 2U_1 - 1$, $V_2 = 2U_2 - 1$,
 $W = V_1^2 + V_2^2$
 - Jika $W > 1$ kembali ke step 1. jika tidak
 $Y = \sqrt{(-2 \ln W)/W}$, $X_1 = V_1 Y$, $X_2 = V_2 Y$



Pembangkitan random variate

- Lognormal
 - Dibangkitkan $Y \sim N(\mu, \sigma^2)$
 - $X = e^Y$
- Beta
 - Dibangkitkan $Y1 \sim \text{gamma}(\alpha_1, 1)$, $Y2 \sim \text{gamma}(\alpha_2, 1)$
 - $X = Y1/(Y1+Y2)$
- Pearson Type V
 - Dibangkitkan $Y \sim \text{gamma}(\alpha, 1/\beta)$
 - $X=1/Y$

