Objectives

• That you understand the necessary attitude:

  >>> Don’t rely on your own impressions. <<<

  < It isn’t natural, it has to be learned hammered in >    < Tullis . . . >

  ➢ Discuss the advance exercise video
  ➢ Suggest what you should rely on.

• That you get a flavour of «Cognitive Systems» Engineering (CSE) in industry:

  ➢ Show an example of CSE applied to an industrial problem
  ➢ Describe what it’s like in practice, how it fits within IT
Outline

• Design example: Context Users Observations Conjecture Design
  – Questions
  – Advance exercise: Discussion Recommandations

• Cognitive Systems Engineering

• Major lessons learned: Methodology and techniques Design

• Articulation with the rest of IT

• Questions (at least 5 minutes at the end)

• Examples of CSE projects (at time permit)

• Support information: Definitions Suggestions Problems Future of CSE References
  Questions during the lecture are OK (in the past, it helped me improve the lecture).
  If needed, I’ll be glad to answer questions after the lecture.

Design example

Alarm annunciator

From diagnosis to situation awareness

(circa 1995, but the lessons remain valid)
Pas de batterie, seulement un peu d’inertie : il faut en tout temps équilibrer la production avec la consommation.

< Si on allume une lumière, il faut qu’un peu plus d’eau coule dans la turbine >
Electrical Power Network: Transmission

- 32,800 km of power lines
- 550 substations

High-voltage power lines
High-voltage substation

Generation
Transmission (high-voltage)

© Hydro-Québec 2017 (André Gascon)

Electrical Power Network: Control

- 1 provincial control center (CCR)
- 7 regional control centers (PA)
- 5 distribution control centers (CED)
- 63 power plants (Generation)
- > 500 substations (Transmission)

CCR: Centre de conduite du réseau
PA: Place d’affaire télé conduite (salle de télé conduite)
CED: Centre d’exploitation de distribution

© Hydro-Québec 2017 (André Gascon)
Electrical network control
< local control room >

Local HMI

< www.hydroquebec.com >

© Hydro-Québec 2017 (André Gascon)

Design example
Context

Electrical network control
< remote control centers >

1 provincial center (CCR)
7 regional control centers (PA)
5 distribution control centers (CED)

Provincial control center HMI (CCR)
Regional control center HMI (PA)

Journal de Montréal

TVA Nouvelles

© Hydro-Québec 2017 (André Gascon)
SCADA: two channels

Process  Automation system  Operator

HMI

Control, statuses

Annunciator

Typical displays (SCADA HMI)

Planning and active monitoring  Control and active monitoring

ADR  SAD  ALB  Annunciator
Annunciator panel

Getting rare, but still in use

( Some automation subsystems can be in use for over 30 years )

Passive monitoring

Chronological annunciator (ALCID-SICC-I)

Getting rare, probably still a few in use

( Some automation subsystems can be in use for over 30 years )

Passive monitoring
Regional control center example (SAD)

Typical annunciator state machine

Field

Alarm

Normal

Alarm message

-

New unack

New ack

Reset

Acknowledge

Return unack

Return ack

© Hydro-Québec 2017 (André Gascon)
Context at Hydro-Québec

• Process control
  – Multiple interdependent variables  - Dynamic
  – Risks  - Conflicting goals
  – Typical task :
    • Information ➔ Situation awareness – Anticipation ➔ Decision

• Hydroelectric power generation and transmission
  – Multiple processes; most of them fairly simple, a few quite complex
  – Complex configuration (and always changing); Complex HMS
  – Expert users; 24/7
  – Important risks; single contingency rule
  – Subject to numerous environmental influences: thunderstorms, freezing rain, wind, temperature, solar activity, rain, river levels, ice, ice cover, forest fires, …

© Hydro-Québec 2017 (André Gascon)

Context at Hydro-Québec

• Expert users
  – Various backgrounds: electrical maintenance, technician, outage coordinator, … (practically nobody with an engineering background)
    • Then local operator, telecontrol operator, telecontrol dispatcher.
  – Long training: master and apprentice, coaching
  – Lots of experience (often at least 10 years of network operation for a provincial control dispatcher).
Context at Hydro-Québec

- To get the right information for design

Situation at Hydro-Québec

- Automation has steadily increased:
  - Number of operators has shrunk dramatically
  - From 20 to 30 (up to 70) substations for a tele-control desk

- Number of alarms points rises sharply with automation:
  - More alarm points than before (from 30 to 300 alarm points for an alternator, upward of 18,000 alarm points in the Beauharnois power plant)
  - Cascades of up to 200 (sometimes up to 400-800) alarm messages at the beginning of a perturbation.

- Network operated much nearer of limits (maximize use of network capabilities)

  ➢ Up to 3000 alarms a day for a tele-control desk.
Advance exercise : Questions

The necessary attitude

“ The best attitude for today's designer to adopt is to think of the user as a different species that one knows nothing about; the user must be investigated scientifically to determine the optimal design features to facilitate the use of interactive systems”

Deborah Mayhew

Be very careful with your own impressions, concentrate on user reactions, try to understand their goals, know the context.

Don’t jump into design until you truly understand the task and the user’s goals.

- Observe the user doing his job
- Analyze the task : understand the user’s goals
- Observe the user using the mock-up or prototype to do his job
The necessary attitude

You are building a tool for somebody who is doing a job you don’t really know about (at least not at the skill level).

On top of that, you know too much about computers to rely on your own impressions to design an HMI.

On top of that, you know too much about computers to rely on your own impressions to design an HMI.

< analogy : driver vs mechanics in car racing >

© Hydro-Québec 2017 (André Gascon)

Observations (alarms)

- After a perturbation :
  - In a power plant, the operators go directly to the control panel, without first looking at the alarm messages on the screen
  - « Acknowledge – reset » without prior reading of alarm messages (the operator looses the historical information, gets the present state)

- High proportion (operators opinion) of one by one judgment about pertinence of alarms
- From 70% to 80% (operators opinion) of alarms are irrelevant
  « 30 alarms (then grouped on 12 indicator lights on the control panel) were enough for an alternator, why do we need 300 now ? »
- In important events, a high proportion of the alarms are momentary
- In some cases, it is difficult to understand the new state of the plant immediately after a complex event. The annunciator doesn’t seem to help.
- Not much pattern recognition for alarms in the regional control center *, only reading of alarm messages

* : except on the mosaic overview display : changes of state (general state of the electrical network)

© Hydro-Québec 2017 (André Gascon)
Techniques

Observation
- At workplace, performing (or simulating) actual work (think aloud), one user at a time
- Notes, screen captures, video camera (with sound), photos

Analysis
- Hierarchical Task Analysis (HTA), strategies, problems, errors
- HTA (goals, data) --> Data, grouping, sequence
- Support situation awareness

Design
- Optimize design with software experts
- Look everywhere for ideas
- Heuristic evaluation

Usability tests
- Use the mock-up to do real task examples (think aloud)
- Users do their task while thinking aloud) (techniques as for observation)
- Iterate

Techniques : Hierarchical Task Analysis

HTA → Information needs → < data, grouping, sequence >
First draft of HMI (interface and task)
- What do you do ?
- When, how often ?
- How do you do it ?
(ask the user to do the actions, or to simulate them)
- What are you looking at ?
- Why do you do it ?
- Problems ? Errors ?
- Strategies ?
- > Look at all the « tools »

During the interview, let the task go (don’t interrupt to obtain direct answers to your interrogations)

© Hydro-Québec 2017 (André Gascon)
Techniques: Hierarchical Task Analysis

In most cases, the tasks already exist in some form

Users can validate the HTA

New HMI → revised task
• main goals mostly intact
• some sub-goals may change
• some actions will probably change
• function allocation (human vs machine) may change

High level understanding from task analysis

• During normal operation
  – Situation awareness / Anticipation
    – Periodic active monitoring
      (period depends of the situation)
  – Mostly passive monitoring of anomalies
    ➢ Alarms managed one by one
Situation awareness

- Perceive
- Comprehend
- Anticipate
  - Anticipation is testable


High level understanding from task analysis

- Perturbation
  - 1 : situation assessment (restoration of lost situation awareness)
    - active gathering of state info; then action on the symptoms
    - annunciator : at most a minor role (even if well designed) *
  - 2 : preparation for return of service
    - annunciator : remaining alarms (« Secondary overview »)
    - decide on a power restoration strategy
    - historical log : may be useful to find causes; then action on the causes

* : a « dark-panel » low-key topographical annunciator is probably part of a better solution
* « unauthorized change of state » indicators in a topographical diagram are useful
Literature and existing products

- Intelligent filtering (to reduce the number of alarms) is difficult to implement and extremely difficult to maintain
- Almost nothing on representation
- No tests with experts
- Situation awareness thought to be important
- Products: design seems neglected (ex.: date-time on the left, including year); looks as if design was done for diagnosis

Conjecture

- After an perturbation, the priority is to re-establish situation awareness; diagnosis is secondary.
- After a perturbation, one needs to update his mental model of the process before making decisions

“The other important aspect of cognitive skills in on-line decision making is that decisions are made within the context of the operator's knowledge of the current state of the process. “

Lisanne Bainbridge 1983
Annunciator Design

First priority after a perturbation: Understand the new state of the plant

HMI has to be designed for the perturbed situations, and must be in constant use *. It must also be useful in normal operation

* : must be already displayed when the event occurs, and the operator must be very familiar with the display.

Techniques: Mock-up

First mock-up
- Paper and PostIt, or Visual Basic used as a drawing tool
- Styrofoam and paper for physical aspects

Evolution
- Usually Visual Basic used as a drawing tool, adding just enough functionality to properly test specific sub-tasks
- Other mock-up tools: PowerPoint, Excel, DHTML (Excel is good for simple web site mock-ups)

Mock-up ≠ Prototype*
- Never (almost) demo the mock-up to users **
- Ask the user to use it to do actual work

The mock-up is the best communication tool between user, client, interaction designer and programmer

Designing the mock-up and doing usability tests enables one to push the interaction design much further

* : a partial prototype may be needed in some cases
** : however, a demo may be useful for clients

© Hydro-Québec 2017 (André Gascon)
Mock-up example (Visual Basic) (SCADA HMI, SAD+A)

State annunciator (mock-up)  (« Liste-État »)
Usability testing (« Liste-État »)

- Camera
- Video
- Commandes de l’annonciateur topographique
- Schémas antérieurs avec annonceur topographique intégré (fonctionnel ou non selon la condition)

3 scénarios
3 scénarios

Visual Substation (commercial version)

© Hydro-Québec 2017 (André Gascon)
ALCID-II : new local annunciator

Design example

Regional control center example

Color rectangles are flashing

Design : mid-1990 (independent, some configuration by HQ)
Currently in use (2007 →)
Provincial control center example

Currently in use (2000→)

Design example

Context

Topographical annunciator (mock-up)

Tests: no advantage, distracting
→ need state first
(alarms are secondary; their representation should remain discreet)

Note on tests: the operators were not familiar with this type of display.

< substation level one-line diagram >
Overview display (« Synoptique Réseau »)

Summer 2007

« I feel the network »

© Hydro-Québec 2017 (André Gascon)

Overview display with alarms (dark-panel, low-key)

Spring 2009

« telecontrol center level one-line diagram »

not done on the substation level one-line diagram (except for CENA)

© Hydro-Québec 2017 (André Gascon)
Annunciators these days

Beauharnois: UCC + control panel

Telecontrol dispatcher desk: PTR + PTS

Provincial control dispatcher desk: several annunciators on monitors (transmission, automatisms, LIMSEL, state estimator) + mimic wall

>>> à compléter (+ dessins ou photos)

• Design example: Context Users Observations Conjecture Design
  – Questions
  – Advance exercise: Context Results Recommendations

• Cognitive Systems Engineering

• Major lessons learned: Methodology and techniques Design

• Articulation with the rest of IT
Overall reasoning

Objective: performance-efficiency of the enterprise in its mission

Means:
- Performance of individuals in their tasks
- Collective performance: business processes, communication

Tools:
- IT tool HMI: information, representation/layout, display HW
- Direct human interactions
- IT services and functions

IT requirements difficult to define for complex systems, especially when users are experts
- Human beings are part of the system
- Tasks
- Processes

N.B.: Standards, guidelines and good practices are very useful, but are far from being sufficient to guarantee a good design.

Specific expertise

Design methodology and techniques, validation techniques

- Cognitive Systems Engineering methodology
  - Integration with requirements engineering and software engineering methodologies.

- Techniques to find the necessary information
  - Interviews, task analysis, usability testing, …
  - < how to get access to the user (intuitive) expertise {in their task} >

- Knowledge of human being: (applied psychology)
  - Vision, perception, situation awareness, decision making, mental load, nature of expertise, human error and biases, representation effect, Gestalt, …

- Display technology, HMI design, interaction design, and business processes description
Demands, needs, design and validation

« In any case, what users want and what users need are two different things, which is why it’s long been a primary usability guideline to watch what users do, rather than listen to what they say. »

« Over the past 25 years, work in usability has shown that one of the best ways to evaluate a design’s quality is by watching users interact with it (through either a functional or mocked-up screen). Again, if years go by before the developers do this, most of their development effort will have been spent producing the wrong design. »

Jakob Nielsen  < www.useit.com >

### Demands, needs and requirements

| What the client asked for | Demands | Description of the design of the business solution, as much as possible validated  
( client/user vocabulary ) |
|---------------------------|---------|----------------------------------------------------------------------------------------------------------------------------------|
| What the client really needs (most of the work of the business analyst is to define those needs). | Needs  
(IT) | Demands ≠ Needs (*) |< High-level business needs > |
| The order given to IT | Requirements  
(IT) | Items of the contract for building/integration/configuration by IT  
( IT vocabulary ) | Requirements = Translation of needs |

* : Give satisfaction to the client ≠ Satisfy all his demands

This is a major success factor. Preferably done via usability testing.
Cognitive Systems Engineering: fundamental concepts

**Design**

*Based on task analysis and usability testing.*

<ie.: based on the users’ expertise in their task>

*With technical optimization.*

*The design is expressed as a mock-up which is also the basic tool for usability testing.*

**Definition of (IT) needs (≠ gathering of needs)**

*Through the design and the validation of the HMI.*

*Interaction design → Functional requirements specification*

---

**Nota bene : Validation vs Verification**

**Validation**

« Confirmation that the product or service, as provided (or as it will be provided), will fulfill its intended use.

In other words, validation ensures that “you built the right thing”. (See also “verification”.) »

**Verification**

« Confirmation that work products properly reflect the requirements specified for them.

In other words, verification ensures that “you built it right”. (See also “validation”.) »

---

© Hydro-Québec 2017 (André Gascon)
Methodology: Main Actors

**Ergonomist**
- Understand the task

**Ergonomist + software experts**
- Design - optimization

**Ergonomist**
- Validate the design

**User expertise**
- Describe their actions

**Users**
- Do their task using the new tool (mock-up)

Methodology: Design Cycle

**Gathering of information on the task** (with users)
- Iterations
  - **Hierarchical Task Analysis (HTA)**
    - **Information needs / sequence**
    - Iterations
      - **Design with optimization** (with computer specialists)
      - **Feasibility analysis**
      - **Mock-up**
      - **Usability tests** (with users)

**Mock-up**
- Iterations
  - **Functional requirements spec.**
    - Definition of mode = Validated design
    - HMI: interaction
    - Services / functions
    - DB

© Hydro-Québec 2017 (André Gascon)
Methodology : « Double Diamond »

- Finding the right problem
  - Explore the fundamental issues
  - Converge upon the real underlying problem.

- Finding the right solution
  - Explore a wide variety of solutions
  - Converge upon a good solution

« Good designers never start by trying to solve the problem given to them : they start by trying to understand what the real issues are. »
« The Design of Everyday Things », Don Norman.

Methodology and design process

- Design process
• Design example: Context Users Observations Conjecture Design
  – Questions
  – Advance exercise: Context Results Recommendations

• Cognitive Systems Engineering

• **Major lessons learned:** Methodology and techniques Design

• Articulation with the rest of IT
• « End »
• Additional information

---

**Major lessons learned: methodology**

• **Demands ≠ Needs** (needs need to be « defined » and validated)

• **Task mode**: to extract information useful for design
  – < task observation, usability testing >

• **Mock-up**: as a design tool, as a medium for usability testing, and as a communication tool
  – < + essential for dynamic aspects, including visual momentum >

• **Main objective**: performance of the human-machine system
  (the performance of the IT system is not an objective per se)

• **Usability testing** and iterations are absolutely necessary

• **Design for situation awareness**: a very powerful concept
Design process

The study of the task (analysis and synthesis) guides the design.

The mock-up supports the creative process, provides a representation necessary to make progress (in the design).

Usability tests enable the designer to validate his design, to uncover design errors, to continue to make progress.

Iterations are necessary because we never get it right the first time.

Objectives are set for the « human-machine system » (performance centered).

Major lessons learned: Methodology

“Most (75%) of the ergonomist’s work is to describe and understand what already exists.”
“... then everything will fall into place…”

Jean-Marc Robert

+ any idea, however brilliant it may look, has to be tested.

Good usability tests are essential.

Good functional (not aesthetics) design is often not noticed by users. They simply won’t complain about the design.
Some lessons learned: Design

• Design for situational awareness

• Design layout for peripheral perception

• Large screens are beneficial for spatial tasks

Design strategies
© Hydro-Québec 2017 (André Gascon)

Some lessons learned: Design

• Design for situational awareness
  - State of the process / data / document . . .
  - State/mode of the application / computer
    ➢ So the user can anticipate correctly
      ➢ Anticipation is testable

• Aware:
  – Structure / « logic » of mental model coherent with application / computer
  – Enough information to ensure that mental model is up to date
  – Applies broadly: aware → anticipate → decision

© Hydro-Québec 2017 (André Gascon)
Some lessons learned: Design [2]

Peripheral perception is important for pattern recognition and navigation

- Design layout for peripheral perception: micro / macro design
  - Know Gestalt and the characteristics of perception via peripheral vision . . .
  - Density not important per se (at least for expert users)
  - « Good » layout is critical
  - Beware of blinking or moving objects

Navigation is a spatial task . . . Visual momentum involves peripheral vision . . .

- Peripheral perception → pattern recognition → situational awareness

Some characteristics of peripheral perception:
- “Black & white”
- Detection of alignments, regularities, Gestalt
- Very sensitive to movement

Some lessons learned: Design [3]

Absolute size of screen is a factor for spatial tasks

Exocentric
(outside of my space)

Egocentric
(within my space)

[ SLAM (body in its environment )
(very robust, very well trained)]

For electrical network diagrams, minimum size ≈ 90’’ diagonal, minimum distance ≈ 80’’

SLAM : Simultaneous Localization and Mapping
• Advance exercise: Context Results Recommendations

• Design example: Context Users Observations Conjecture Design

• Cognitive Systems Engineering

• Major lessons learned: Methodology and techniques Design

• Articulation with the rest of IT

---

Business Analysis and Cognitive Systems Engineering

Business solution = business process (and tasks)  
Bottom-up analysis: from the individual tasks

Gather / Inquire
- Objectives
- High level requirements
- Business rules

Interviews / observation
- Task analysis
- Strategies/Errors/Problems
- Information

Inquire
- Current data

Identify problems
- Final causes

Model the current business process

Gather / Inquire

Model the future business process

Future business solution

Model the future tasks

Design and validate HMI

Describe UC

Define future data

Requirements specification

Business solution

IT solution

EPC: Event (driven) Process Chain

© Hydro-Québec 2017 (André Gascon)
« Un pont entre les Affaires et les TI » (CS et AA)

**Analyse d'affaires**

- Gestion client
- Fiches projet
  - Suivi TI (projets et DT)
  - Plan d'affaires
- Processus d'affaires
  - Business Case
- Problématiques
  - Intrants aux fiches projet
  - Demande / besoin
  - Intrants au Business Case
- Essais
  - Analyse de tâche
  - Essais d'utilisabilité

**Conseiller stratégique**

**TI**

- Portefeuille de projets TI
- Portefeuille de maintenance
- Avis de conformité
- Bureau d'architecture
- Priorisation projets (via fiches projet)
- Architecture-cible
- Bilan de santé

**Projets**

- Comité de coordination
- Accompannement
- Architecture projet
- Plan de projet / Recommandation
- Avis de conformité
- Analyse préliminaire
- Réalisation - relation réalisateur
- Essais d'acceptation
- Business Case (longitudinal)

**Portefeuille de projets TI**

- Évolution
- Maintien
- Corrective
- Pérennité
- Adaptative demandant code
- Carnet de maintenance
- Demande de transformation
- Priorisation DT / calendrier liv.

**Portefeuille de maintenance**

- Évolution
- Maintien
- Corrective
- Pérennité
- Adaptative demandant code
- Carnet de maintenance
- Demande de transformation
- Priorisation DT / calendrier liv.

Vert : CS impliqué dans l'activité
Bleu : AA impliqué dans l'activité

**Element of a successful IT project**

*Four essential ingredients* for a successful project
- Business process analysis, strategic alignment (, …)
- Interaction design (display devices, appearance et behaviour)
- Technical design
- Project management

**Definition of needs** (funcionnal)
- Demands ≠ needs
- Needs = Interaction design
- It is essential to validate the design
- Tasks and processes

Task observation and analysis is the foundation for both interaction design and business processes analysis

< Interactive systems >

< Objective = profitability (max perf/cost ratio) >
Role of Cognitive Systems Engineering

CSE = more than the Human-Machine Interface (HMI)

Role of Cognitive Systems Engineering

CSE: insure tool will improve the performance of the HMS

SE: insure technical quality, control costs and delays

Requirements specification

Task (and business processes) design Functions, services, data

> Help the clients define their needs <

© Hydro-Québec 2017 (André Gascon)
Articulation with « the rest of IT »

BA - UX  Business Analysis
User Experience

Corporate objectives
Corporate strategies
Corporate structure

Business process analysis
Task analysis
Interaction

Functions (services) / data

Design levels

Corporate needs (high-level)

Operational design
Interaction design
Functional design

Requirements
Constraints / Ideas

BA
UX

© Hydro-Québec 2017 (André Gascon)
Éléments fondamentaux de LEAN

**Obsession continue**
- Minimiser le gaspillage (ce qui ne produit pas de valeur pour le client)
- Maximiser la valeur pour le client
- Régularité du flux : stabilité de la capacité et de l'expertise

Cette valeur est elle-même subordonnée à la valeur de l'activité (supportée par l'outil TI) pour les clients de TransÉnergie et d'Hydro-Québec (entreprise)

Flux tirés (« Pull ») : chaîne de « clients » internes aux TI, jusqu'au client de TI lui-même; puis jusqu'aux clients de TransÉnergie et d'Hydro-Québec.

**Fondations : terrain (« Gemba »), et analyse de tâche**

CSE vs frameworks and books of knowledge

“Frameworks” and “Books of Knowledge” are usually meant to accommodate any design methodology.

One must first choose a design methodology to make relevant choices in a framework or a book of knowledge.

For software with a human interaction aspect, the best design methodology is probably the “user centered design” approach of the ISO-9241-210 standard (ex-13407).

[ CMMI, Macroscope, Agile, BABoK, PMBok, SwEBoK, ... ]
### Some references

- « CITATION.doc »

---

### Some references

End

Questions?

Gascon.Andre@hydro.qc.ca

Additional information

- Mock-up examples
- Cognitive Systems Engineering definition
- More on lessons learned
- Difficulties
- Suggestions
- My opinion on the future of Cognitive Systems Engineering
- More on process control
- More context information for advance exercise
- How I got to CSE
- Requirements specification table of contents
- Industrial engineering and CSE

Mock-up examples

→ ../ICS/Pres-ICS-IdT-2005/Exemples-Realisation-ICS.ppt

3, 5 : Beauharnois
8 : ALCID-II
12, 13 : SEQAM
14 : ADR
17 : étiquetage
19 : revue Transport CCR
20 : procédure aidée par ordinateur
21, 22 : délestage cyclique
28 : DREX
35 : pupitres CCR
39, 40, 41, 42 : synoptique réseau
44, 45, 46 : PG&E

© Hydro-Québec 2014
© Hydro-Québec 2017 (André Gascon)
Cognitive Systems Engineering

• « Discipline that :
  – Takes an interest in all stages of the life of complex human-machines systems (« Cognitive Systems »)
  – Calls for knowledge and methods of many nature, social and human science disciplines
  – Has the fundamental objective to improve the performance of human-machine systems »

Jean-Marc Robert

Lessons learned

• Analyze work activities (and use HTA to design the first draft of the HMI)
• Don’t consider demands as needs; try instead to uncover the real needs
  – Beware of needs expressed as solutions
• Insure that the user is in « task mode » to obtain the right information
• Don’t ask users to validate the requirements specification
• Beware of the pitfalls of participative design
• Keep contact with the field by regular visits
• Evaluate the HMI according to the task logic and the performance of the human-machine system
• Do not aim for user-friendliness as a primary objective
• Build mock-ups
• Do the complete Cognitive Systems Engineering design cycle
• Explain the design process to the decision makers to convince them of the merits of the recommended design.
  One way to help convince project leaders, expert consultants, client representatives, …. is to tell stories about users performing their tasks. Seeing a video of users in their activities may also be useful.
  – describe the main problems encountered by the users and their consequences ($) 
  – describe observations and the actual design process

© Hydro-Québec 2017 (André Gascon)
Difficulties

- Solutions becoming needs, or needs expressed in the form of solutions
  - Technical « trips » : solutions in search of a problem
- CSE results often difficult to explain to software developers (in fact to anybody who didn’t spend much time observing users at work); also often difficult with ex-users : they don’t feel the need to check with users, they stick with their own impressions.
- Many computer specialists rely on their own impressions, and think they can design good HMI.
- Most managers don’t see the importance of a good HMI design (enabling performance of users in their tasks, not only « user-friendly »), or of a bigger monitor, for that matter.
- Difficult to convince project leaders to accept the time and cost of CSE
- Usability testing limits with experts

Suggestions

- When a client demands something, find out why, find out what are the task objectives
- Pay regular visits to users; be also there when the software is installed
  → Stay « grounded » in the reality of the field
- Always check with users (3-4 at least, if possible); but don’t let them do the whole design; don’t rely on your own opinion
- Never (almost) do demos to users, instead ask them to use the product to do actual work
- Be rigorous (do a complete CSE cycle), not necessarily formal

- < citations.doc >
CSE future (my opinion)

Compatibility with human being
Better representations: better adapted to human beings
Better interaction means: better adapted to human beings
User aware of what the computer is doing  {Human-Computer Cooperation (Norman)}

Compatibility with task
Still far from complete generic solutions ➔ bottom-up analysis is essential
Importance of details    Nature of expertise

Articulation with business processes and the structure of the organization

Business Analysis

HMI design for process control: future (my opinion)

Process Control
Tasks
95% = acquisition of information, judgment on the quality of the information; importance of presentation
5%   = decision

Monitoring ➔ better situation awareness, without saturation (especially the rapid restoration of situation awareness following a complex event)
( up to date mental model of the « process + computer » )

➔ ? Design for peripheral perception? ❯
Advance exercise: some context information

Alarms: part that is not automated

Capture of momentary alarms
Alarm management
("son", "acq", "rap")
(2 steps not to miss alarms)

- Undo not available
- Locate
- Mock-up is not perfect

Dimly lit room, dark panel
Other windows = color on black

Separate « New » and « Return to normal »
messages useful in log (for analysis after the fact)
< but not in real time >

How I got to Cognitive Systems Engineering

- Physicist

- Programmer-analyst (real-time software)
- Project leader (automation software)

- Requirements specification, automation of the Beauharnois hydroelectric power plant
- Cognitive Systems Engineering to define (IT) needs
How I got to Cognitive Systems Engineering

SICC-I console (no longer in use)

Sections du T110 et documents P+

1. Introduction
1.1 Portée du document
1.2 Audience visée
1.3 Portée de l’outil < P140 >
1.3.1 Résumé des fonctions < P250 >
1.3.2 Applications de l’outil < P140 ? >
1.4 Définitions, acronymes et abréviations
1.4.1 Termes et acronymes spécifiques à Hydro-Québec
1.4.2 Termes et acronymes d’usage courant
1.5 Références
1.6 Vue d’ensemble du document
Sections du T110 et documents P+

2. Vue d'ensemble de la solution d'affaires
   2.1 Contexte du système / projet < P140 >
   2.1.1 Raison d'être du système < P140 >
   2.1.2 Situation actuelle < P120 >
   2.1.3 Demandes < P100 >
   2.1.4 Sources d'exigences
   2.1.5 Présomptions et dépendances
   2.2 Description générale de la solution d'affaires < P140 > < P240 >
   2.2.1 Objectifs du système humains-machines < P140 >
   2.2.2 Relations à l'entreprise et aux processus d'affaires < P140 >
   2.2.3 Interaction humain-machine < P240 >
   2.2.4 Interfaces < P240 >
   2.2.5 Fonctions majeures < P250 >
   2.2.6 Structure des données < P170 >
   2.2.7 Contraintes < P261 >

Sections du T110 et documents P+

3. Exigences spécifiques
   3.1 Interfaces externes
   3.1.1 Interfaces système < P >
   3.1.2 Interfaces utilisateurs < P240 >
   3.1.3 Interfaces matérielles < P261 >
   3.1.4 Interfaces logicielles < P261 >
   3.1.5 Interfaces de communication < P261 >
   3.2 Fonctions < P240 > < P250 >
   3.2.1 Fonction …
   3.3 Performance < P240 >
   3.3.1 Capacité (statique)
   3.3.2 Performance dynamique
   3.4 Base de données < P240 >
   3.5 Contraintes < P261 >
Sections du T110 et documents P+

3.6 Attributs du système TI
   3.6.1 Fiabilité < P240 > < P261 >
   3.6.2 Disponibilité < P240 > < P261 >
   3.6.3 Sécurité < P240 > < P261 >
   3.6.4 Maintenabilité < P240 > < P261 >
   3.6.5 Adaptabilité < P261 >
   3.6.6 Pérennité < P261 >
   3.6.7 Portabilité < P261 >
   3.6.8 Compatibilité < P261 >
   3.6.9 Autre attribut …

3.7 Autres exigences
   3.7.1 Structure de l’information < P170 >
   3.7.2 Architecture logicielle < P261 >
   3.7.3 Structure des traitements et interfaces < P250 >
   3.7.4 Stratégie d’implantation < P261 >
   3.7.5 Stratégie de changement organisationnel < P140 >
   3.7.6 Impacts < P140 >
   3.7.7 Coûts et bénéfices < P140 >
   3.7.8 Infrastructure technologique < P261 >
   3.7.9 Langue < P240 >
   3.7.10 Support < P261 >
   3.7.11 Autre exigence particulière … (efficience, …)

3.8 Groupes d’essais < P410 >

Sections du T110 et documents P+

4. Annexe : Registre des raisons de conception
5. Annexe : Essais fonctionnels
6. Annexe : Essais de performance et de limites
7. Index
Cognitive Systems Engineering

• Industrial engineering = performance and efficiency
  – Better tools ➔ better performance in task (speed, errors, quality, …)
  – Applied to software (considered as a tool) = performance and efficiency of the Human-machine system (HMS)
  – Applied psychology: because a human being is part of the system