4. Families of Systems

Overview of SCS4, Again

1. *rigorous description* of requirements
2. *what information* should be provided in computer system documentation?
3. *decomposition* into modules
4. *families* of systems

Overview of Chapter 4: Families of Systems

4.1 motivation:
maintenance problems in telephone switching

4.2 families of programs

4.3 families of requirements

4.1 Motivation: Maintenance Problems in Telephone Switching

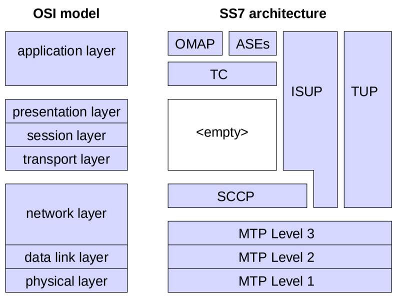
Overview of Chapter 4.1

- background on telephone switching
- feature interaction problems in telephone switching

History of Telephone Switching Systems

. . .	. . .
1950s	direct distance dialling (DDD) No. 5 Crossbar
early 1960s	stored program control switches
1976	Signalling System No. 6
1980	*Signalling System No. 7*
1984	*ISDN*
currently	IP telephony

Signalling System No. 7



OMAP: Operations, Maintenance and Administration Part
ASE: Application Service Element
TC: Transactions Capabilities
ISUP: ISDN User Part
SCCP: Signalling Connection Control Part
MTP: Message Transfer Part
TUP: Telephone User Part

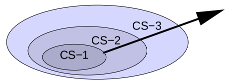
ISDN/DSS1

- Integrated Services Digital Network
- basic service:
  - two B-channels (64 kbit/s, transparent)
  - one D-channel (16 kbit/s, for signalling, e.g., call setup)
    - ▷ protocol: Digital Subscriber Signalling 1 (DSS1)
- supplementary services:
  - Calling Line Identification Presentation
  - Call Forwarding
  - Closed User Group
  - User-to-User Signalling
  - . . .

- fixed set of supplementary services

Intelligent Network (IN)

- extension of telephone switching systems
- general goals:
  - ○ rapid introduction of new services
  - ○ broaden range of services
  - ○ multi-vendor environment
  - ○ evolve from (all) existing networks
- standardized by ITU-T
- approach: base service & additional services/features
- new services step by step:

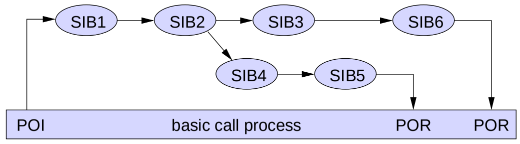


Intelligent Network Conceptual Model (INCM)

- four "levels":
  - ○ service plane
  - ○ global functional plane
  - ○ distributed functional plane
  - ○ physical plane

Global Functional Plane

- service independent building blocks (SIBs)
- service logic ("glue" for SIBs)
- basic call process
  - ○ is special SIB
  - ○ POI: point of initiation (of service)
  - ○ POR: point of return



Services in IN CS-1

- Abbreviated dialling
- Account card calling
- Automatic alternative billing
- Call distribution
- Call forwarding
- Call rerouting distribution
- Completion of call to busy subscriber
- Conference calling
- Credit card calling
- Destination call routing
- Follow-me diversion
- Freephone
- Malicious call identification
- Mass calling
- Originating call screening
- Premium rate
- Security screening
- Selective call forward on busy / don't answer
- Split charging
- Televoting
- Terminating call screening
- Universal access number
- Universal personal telecommunications
- User-defined routing
- Virtual private network

- 25 services
- kind of services limited:
  - ○ mainly for call setup and call tear down
  - ○ 1 customer and 1 call leg only, mostly
- set is "political":
  - ○ some services very similar
    - ▷ taken from different sources, without proper merge
    - ▷ example: Televoting / Mass Calling

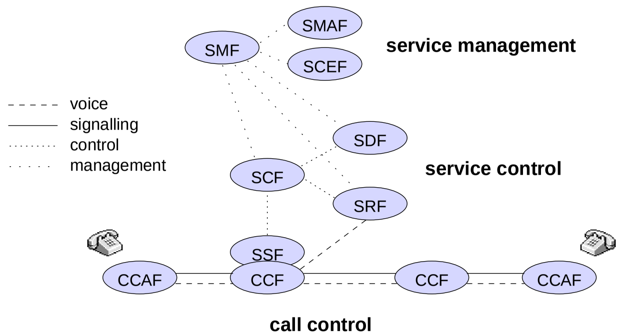
Features in IN CS-1

- Abbreviated dialling
- Attendant
- Authentication
- Authorization code
- Automatic call back
- Call distribution
- Call forwarding
- Call forwarding on BY/DA
- Call gapping
- Call hold with announcement
- Call limiter
- Call logging
- Call queueing
- Call transfer
- Call waiting
- Closed user group
- Consulation calling
- Customer profile management
- Customized recorded announcement
- Customized ringing
- Destinating user prompter
- Follow-me diversion
- Mass calling
- Meet-me conference
- Multi-way calling
- Off net access
- Off net calling
- One number

- Origin dependent routing
- Originating call screening
- Originating user prompter
- Personal numbering
- Premium charging
- Private numbering plan
- Reverse charging
- Split charging
- Terminating call screening
- Time dependent routing

- 38 features

Architecture of Distributed Functional Plane



Basic Call State Model

- originating BCSM
- terminating BCSM

Feature Interaction Problems in Telephone Switching

- *features work separately, but not together*
  - hundreds of (proprietary) features
  - combinations cannot be checked anymore
- telephone switching
  - users' expectation high
- feature
  - about any increment of functionality

Calling Card & Voice Mail

- #-button
  - (Bell) calling card:
    start new call without re-authorization
  - (Meridian) voice mail:
    end of mailbox number, end of password, . . .
- call voice mailbox using calling card??
  - either early disconnect, or
  - calling card feature crippled
- resolution by Bell
  - introduce new signal:
    "#-button pressed at least 2 sec."

Call Waiting & Call Forward on Busy

- both activated simultaneously
  - in busy state
  - when another call arrives
- only one can get control
  - no resolution, except restrictions on features

Originating Call Screening & Area Number Calling

- OCS
  - aborts calls to numbers in list
  - query Service Data Point (SDP) for list
- ANC
  - dialled number + area(calling number) → called number
  - example: Domino's Pizza
  - query SDP for called number

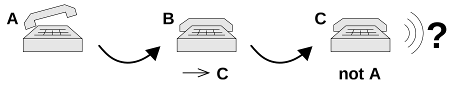
- switch may restrict no. of queries
  - protection against infinite loops
  - e.g., one query per call
  - → OCS subscription prevents orders for pizza
- solution: one more query??

Calling Number Delivery & Unlisted Number

- conflict of goals
  - CND reveals caller
  - UN prevents revealing caller
- resolution
  - weaken one feature
  - e.g.: CND delivers only 1-111-1111-1111
    for unlisted number

Call Forwarding & Terminating Call Screening

- CF
  - B forwards all calls to C
- TCS
  - when A is caller, C blocks him
- A calls B: can/should A reach C?



- notion of "caller" is crucial

Informal Feature Interaction Definition in Literature

- *FI:*
  the behaviour of a feature is changed by another feature
- not precisely clear what a feature actually is
- not all interactions are undesired

Categorization of Causes

according to [Cameron et. al.]:

- violation of feature assumptions
  - naming
  - data availability
  - administrative domain
  - call control
  - signalling protocol
- limitations on network support
  - limited CPE signalling capabilities
  - limited functionalities for communications among network components

- intrinsic problems in distributed systems
  - resource contention
  - personalized instantiation
  - timing and race conditions
  - distributed support of features
  - non-atomic operations

Approaches for Tackling FI

- ignore
- informal
  - filtering
  - heuristics
  - . . .
- formal methods
  - validation of:
    - ▷ specified properties of the features
    - ▷ general properties of the system (free of non-determinism, . . .)

- new architectures
  - IN
  - Tina, Race, Acts
  - DFC, agents
- better software engineering processes

- in practice: ignore / informal / processes / (architectures)
- formal analysis?
  yes, but. . .
  - formalization is huge task
  - complexity not amenable to tools
    - ▷ "spaghetti code" dependences

Feature Interactions in the Requirements

- if requirements complete,
  all FI are (inherently) present in the requirements

Requirements Structuring Problems

- monolithic requirements or single layer of extension
  - ISDN: monolithic
  - IN: no features on top of features
  - CF & TCS: resolution needs extended, common notion of caller
  - CF & OCS: resolution needs extended, common notion of called user

- new services depend implicitly on new concepts
  - some new concepts:
    - ▷ conditional call setup blocking
    - ▷ dialled number translation
    - ▷ multi-party call/session
      - required for CF & TCS and for CF & OCS
    - ▷ service session without communication session
    - ▷ distinction user – terminal device
    - ▷ distinction user – subscriber
    - ▷ mobility of users and of terminals
      - difficult to specify with network of distributed switches
    - ▷ multiple service providers, billing separately

- concerns of the users' interface are spread out
  - several features asume exclusive access to the user's terminal device (12 buttons + hook)
  - example: calling card & voice mail

Needed: a More Modular Requirements Structure

- centralize responsibility for the users' interface
- a layered architecture
  - like in computer communication systems

New Architectures

- current: IN
  - currently largest impact on implementations
    - ▷ see above
  - Jain
    - ▷ enhanced IN-like architecture
    - ▷ developed currently
    - ▷ in Java
    - ▷ allows multi-party, multi-media calls
    - ▷ Java Call Control (JCC):
      call state machine similar to that of the IN
    - ▷ JCC does not handle feature interactions

- future: Tina, Race, and Acts
  - Tina
    - ▷ radical approach: entirely new architecture
    - ▷ strongly based on Open Distributed Processing (ODP) and Corba
    - ▷ migration difficult
  - Race project
    - ▷ Cassiopeia
      - developed open services architectural framework (Osa)
      - many commonalities with Tina
      - focuses on requirements engineering of services
      - tries to take legacy services into account
    - ▷ Score
      - concerned with the methodological aspects of service creation
      - detection of undesired service interactions:
        formal methods, exhaustive simulation
        applied to small example

- Acts project
  - ▷ followed Race project
  - ▷ application and on evaluation of service architectures
  - ▷ result: a modified architecture

- research: the DFC and the agent architecture
  - Distributed Feature Composition (DFC)
    - ▷ compose features in a pipe-and-filter network
    - ▷ designed to be implementable on a conventional switch
    - ▷ some new concepts supported, others not
    - ▷ no layered architecture
    - ▷ implemented in AT&T's Eclipse project,
      which additionally incorporates Voice Over IP
  - Zibman et. al.'s agent architecture
    - ▷ separates several concerns explicitly
    - ▷ restricts itself to narrow-band telephony over a fixed network
    - ▷ Plain Old Telephone Service is represented by a single service agent

Discussion of New Architectures

- IN important step, but not sufficient
- Tina, Race, Acts have most of the interesting concepts, but transition is very expensive
- feature interaction detection is still research

- some undesired service interactions still possible in new architectures
  - a paper checked the FI benchmark for Tina
  - still possible:
    - ▷ forwarding loop
    - ▷ automatic callback & automatic re-call
    - ▷ calling number delivery & calling number delivery blocking
    - ▷ billing problems for video conference
    - ▷ . . .
  - causes: violated assumptions or conflicting goals
- *how to prepare for unanticipated changes??*
  - at least encapsulate as much as possible