Framework and Requirements for the Internet Intelligent Networks

L. Slutsman, G. Ash

AT&T Labs

F. Haerens

Alcatel

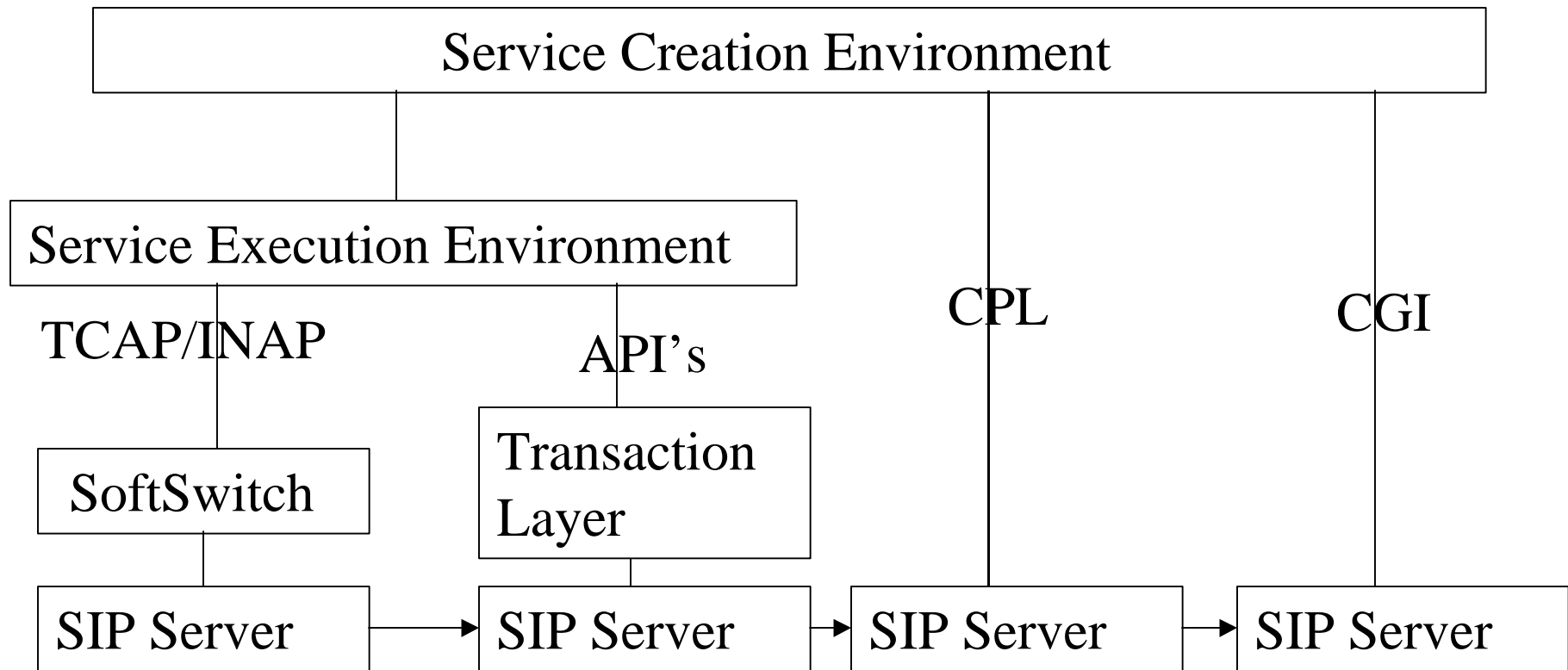V. Gurbani

Lucent Technologies

Outline

- Generalized IIN architecture
- Transparent Access from SIP networks to traditional IN Services
- Internet-Service-Creation (CPL, CGI)
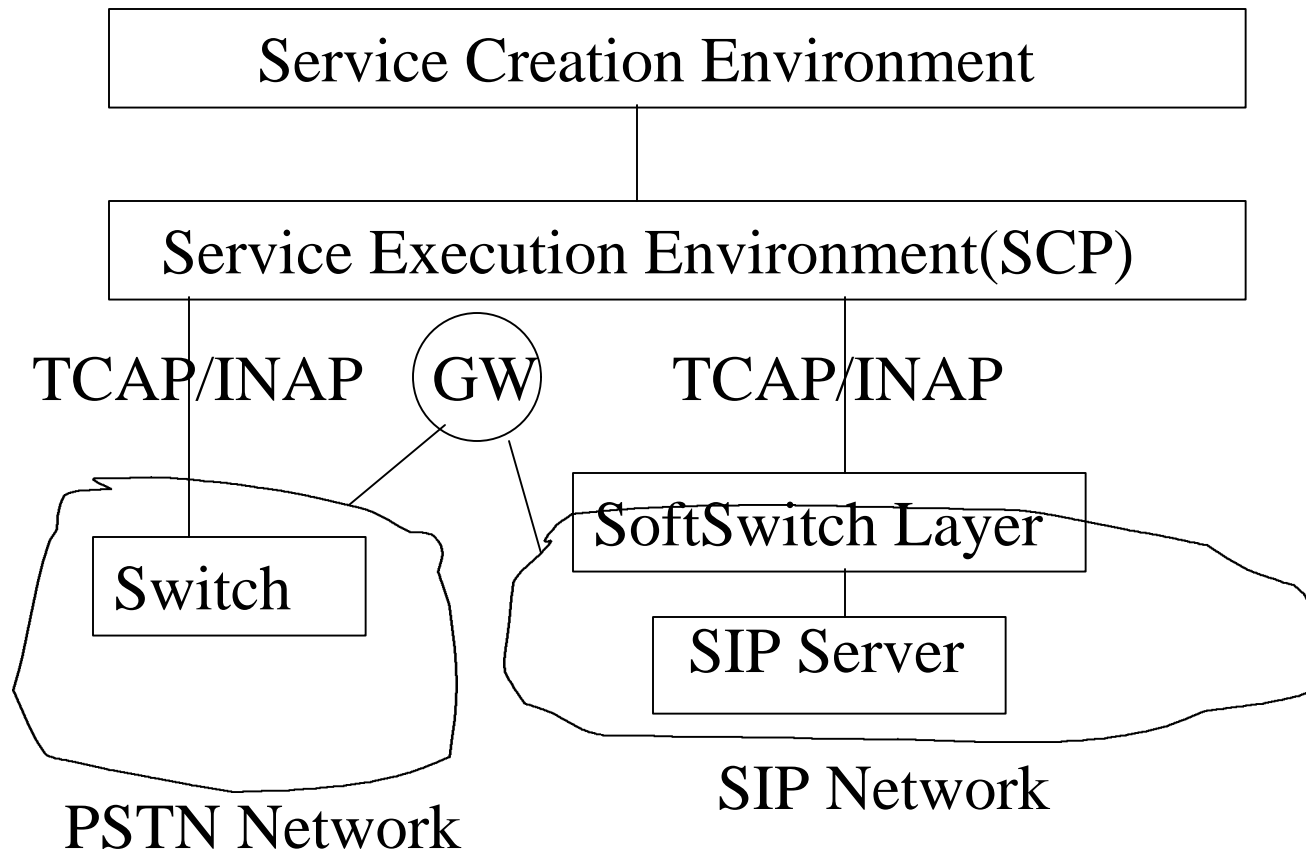- API Access to Service-Creation
- Service Example

Transparent Access to Traditional IN, Internet-based Service Creation, & APIs

- Motivation:
  - PSTN and VoIP networks will coexist for some time;
  - reuse of hardware (SCP,SN,SMS), software, and service logic
  - time-to-market for introduction of VoIP services;
  - propose to include access to traditional-IN together with internet-based service creation (CPL, CGI)
  - API technologies (e.g. Parlay, Jain) also may be incorporated

Generalize IIN architecture

Service Creation Environment

Service Execution Environment

TCAP/INAP

API's

CPL

CGI

SoftSwitch

Transaction Layer

SIP Server	→	SIP Server	→	SIP Server	→	SIP Server

Traditional IN Architecture

Service Creation Environment

Service Execution Environment(SCP)

TCAP/INAP GW TCAP/INAP

Switch

SoftSwitch Layer

SIP Server

PSTN Network

SIP Network

Architecture (Cont)

- Based on Remote Execution of Service Logic.

- SoftSwitch Layer makes SCP believes that it deals with "switches".

- Challenge is to map SIP FSMs into IN BCSM.

- Limitations:1) heavily relies on SS7; 2) performance degradation due to remote execution.

Internet Based Service Creation

- The execution of the service logic takes place on the server.

- Call Processing Language (CPL)-- condition-action pairs based language.

- Common Gateway Interface (CGI) for SIP:
  – programming language independent;
  – derived from CGI-HTTP

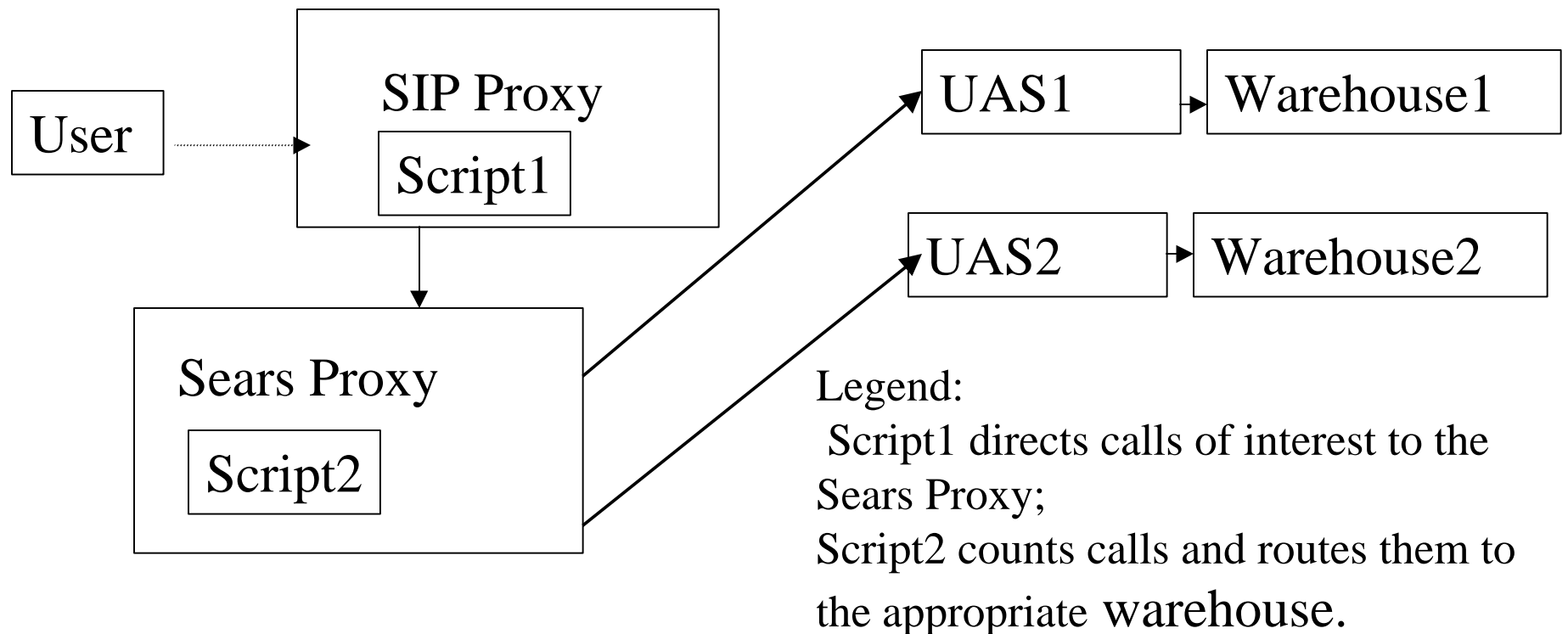- Others

Generalize IIN architecture (Cont)

- How to Invoke a service logic (e.g. script):
  - Use a trigger database: ("Trigger",P_1,..P_n), where Pi is the ordered list of service programs; "Trigger"=("SIP_Call_Leg", S_i) and S_i is a state from the corresponding SIP transition diagram.
  - In addition, information that comes with message, along with "global" variables are used by service logic.

Service Example

- Service: calls directed to to catalogorder@sears.org should be distributed between two warehouses in ratio 3:5.
- **"Trigger":**
  - **To=catalogorder@sears.org**
  - **From=***
  - **Call_ID=***
  - **SIP State=S_1**

The Distributed Service Execution Environment

User		SIP Proxy

SIP Proxy contains: **Script1**

Sears Proxy contains: **Script2**

UAS1 → Warehouse1

UAS2 → Warehouse2

Legend:
 Script1 directs calls of interest to the Sears Proxy;
Script2 counts calls and routes them to the appropriate warehouse.

Centralized Service Execution Environment(SCP)

SCP: updates number of calls to each warehouse and determines the destination

Query   Destination   UAS1   Warehouse1

SoftSwitch

User   SIP Proxy   UAS2   Warehouse2