

VCIS

Voice Case Information System

by

Selim Mimaroglu

The background of the slide is a solid blue color. In the lower right quadrant, there are several concentric, light blue circular ripples that resemble water droplets hitting a surface, creating a decorative pattern.

Grant Project

- I did Voice Case Information System (VCIS) for the United States Bankruptcy Court
- This is a grant project
- This is the only grant project completed successfully in the history of United States Bankruptcy Court (nationwide; 300 courts)

What is it?

- VCIS: is an Interactive Voice Response Telephony Application System
- It gives information about:
 - Debtor Name
 - Filing date of the case
 - Attorney of Debtors
 - Trustee
 - Judge
 - Meeting of creditors
 - Case information: closed? discharged? open?
 - Party names spelled
 - etc

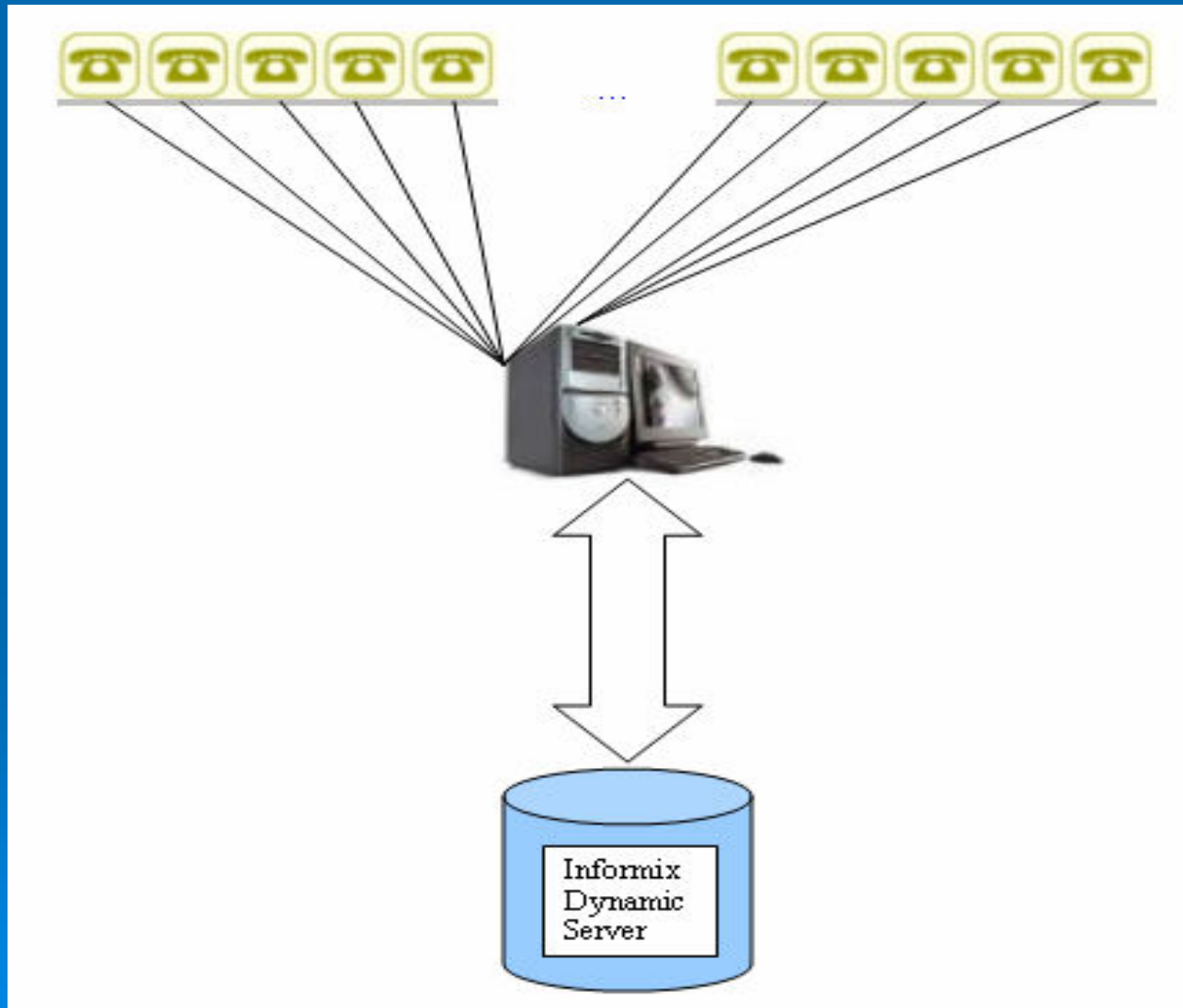
Awards

- 1) “Award of Excellence”
 - 2) Cash Award
- They don’t give these awards easily, especially the first one

Software Life

- It's been running for a few months
- I am adding new features
- It will be distributed to 300 courts nationwide (some courts started to use it)

Overview of the System



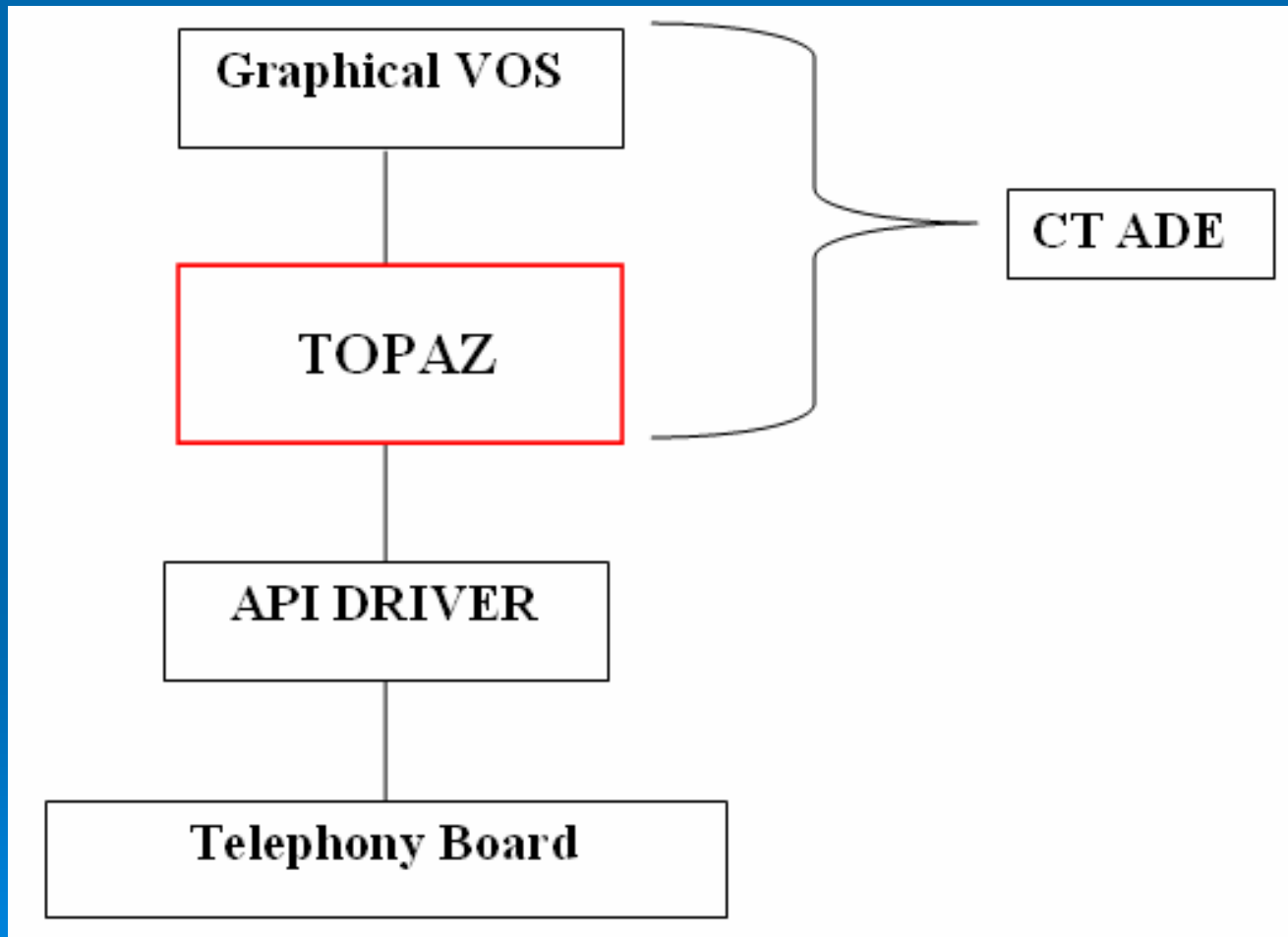
Hardware

- Intel D/120JCT-LS *This is a 12 port telephony board*
- Intel CTADE2DEVKEY *This is a development key for 2 ports*
- Intel CTADE12RT *This is a 12 ports runtime key*
- Intel/Dialogic Breakout Box
- Intel/Dialogic Breakout Cable
- x86-based PC (CPU: 1.7 GHz, Memory: 2GB)

Software

- Intel Computer Telephony Application Development Environment (CT ADE)
- Rhetorical Text to Speech Engine, rvoice
- Operating System: Windows 2000
- Microsoft SAPI5 (Speech API)
- Microsoft IIS (Internet Information Systems)
- Intel Dialogic System Software (Driver)

Architecture of CT ADE (simplified)



Graphical VOS

- VOS Language Compiler and Runtime Engine
- Flow Charter
- Editor
- Debugger

Topaz

- Topaz provides a level of abstraction
- There are too many Telephony boards, you don't have to know all the details
- Topaz will work with "any" telephony board (I know it works fine with Intel's D/120JCT-LS)
- It's possible to directly code to the driver in C/C++
- Topaz makes it easier, it handles the details

Topaz Scanner

- Finds out all resources on a system, and corresponding drivers such as; Telephony board, Text to Speech Engine
- It's flexible
- You can manually add resources

Resources in Topaz

- **Trunk Resources:** are responsible for call control. Call control includes dialing out, accepting an incoming call, and hanging up when a call is finished
- **Media Resources:** Control the playing and recording of sound files and tones as well as getting DTMF (Dual Tone Multi-Frequency) digits from callers

Resources in Topaz (cont)

- **Fax Resources:** Controls the transmission and processing of fax data
- **Voice Recognition Resources:** translate a caller's spoken input into text strings.
- **Text to Speech Resources:** translates text strings into spoken output.
- **Conferencing Resources**

VOS Language Concepts

- It's a C like language
- No memory management
- Case sensitive
- Loosely typed
- Max variable length is 127
- Can form complex expressions
- Many built-in functions

VOS Language Concepts (cont)

- switch/case
- goto
- include
- for loop
- while loop
- do...until loop
- user defined functions

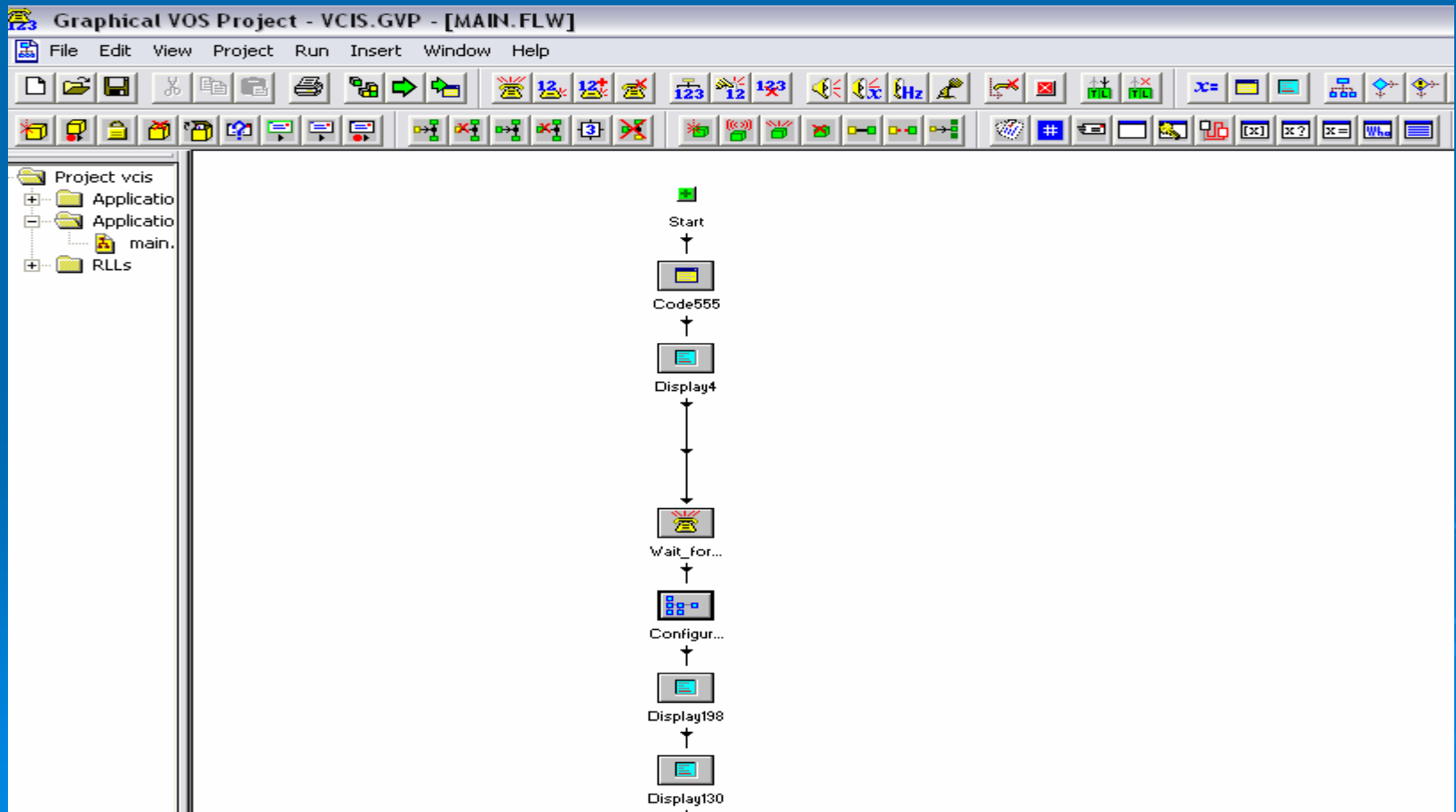
Code Snippet

```
#
# Loop_Start18 - Loop Start Cell
#
Loop_Start18:
$if (DEBUG streq 1)
    voslog("@L Enter Cell::Loop_Start18 - Loop Start Cell on Trunk=" & sysPhoneLine);
$endif
Start_Loop_Start18:
    if (not(i < length(firstAttorneyPhoneNo)))
        goto Stop_Loop_Next18;
    endif
$if (DEBUG streq 1)
    voslog("@L Exit Cell::Loop_Start18 - Loop Start Cell on Trunk=" & sysPhoneLine);
$endif
    goto Code394;

#
# Loop_Next18 - Loop Next Cell
#
Loop_Next18:
$if (DEBUG streq 1)
    voslog("@L Enter Cell::Loop_Next18 - Loop Next Cell on Trunk=" & sysPhoneLine);
$endif
    goto Start_Loop_Start18;
Stop_Loop_Next18:
$if (DEBUG streq 1)
    voslog("@L Exit Cell::Loop_Next18 - Loop Next Cell on Trunk=" & sysPhoneLine);
$endif
    goto Code395;
```

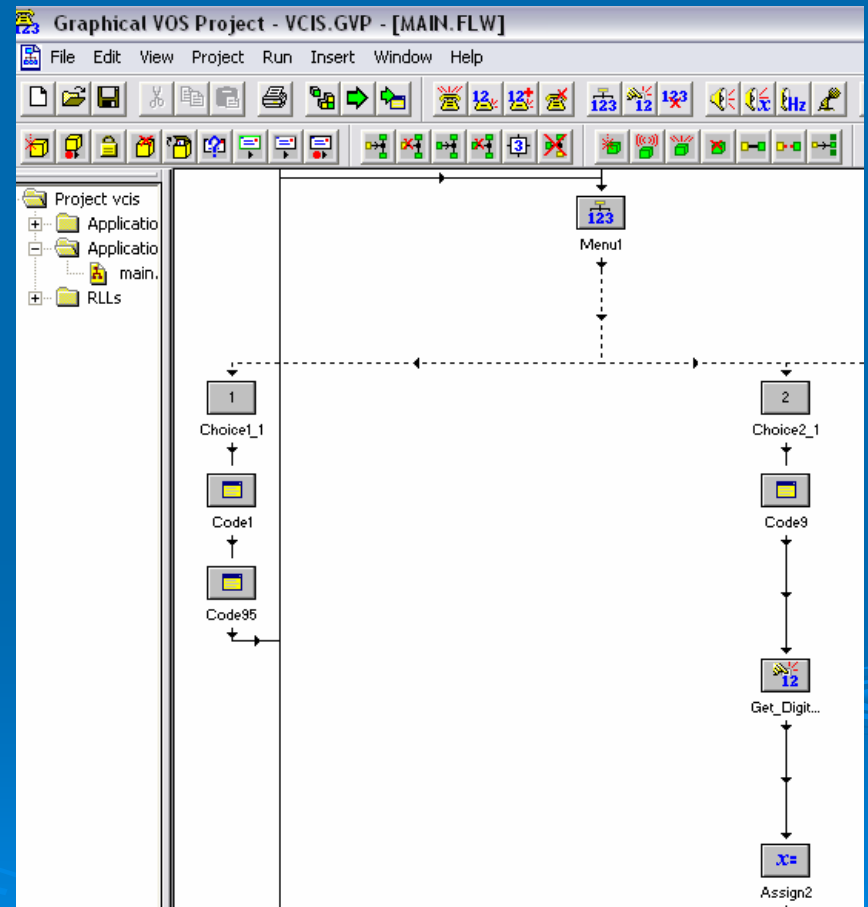
Graphical VOS

➤ is something like Visual Studio .Net



How it works?

- Drag & drop cells you want to use. You can add code as well.
- Total 16,000 lines; some generated by system some by me.
- First priority is error handling
- It's capable of handling 17,280 calls per day (average 1 cal/min)



Improvements

- Better hardware: Sun Machine instead of PC
- Better OS: Unix instead of Windows 2000
- Better Language: Java instead of VOS
- Unfortunately the board doesn't support any of the above

Future Work

- Support for the system
- Some modifications for higher level courts; such as Appellate Courts
- I would like to have a grant proposal on “Wireless Web”
- Most of the cell phones, smart phones are able to handle this technology.