



---

**Poseidon House  
Castle Park  
Cambridge CB3 0RD  
United Kingdom**

TELEPHONE:  
INTERNATIONAL:  
FAX:  
E-MAIL:

**Cambridge (0223) 323010  
+44 223 323010  
+44 223 359779  
apm@ansa.co.uk**

---

## **ANSA Phase III**

# **Tools (Nov. 93 TC presentation slides)**

**Rob van der Linden**

### **Abstract**

This file contains the slides for the Tools presentation for the November 1993 TC presentation.

APM.1058 (ANSA and Tools) accompanies the slides in this document.

---

APM.1085.00.01

**Draft**

29 October 1993

Request for Comments (confidential to ANSA consortium for 2 years)

---

**Distribution:**

**Supersedes:**

**Superseded by:**

Copyright © 1993 Architecture Projects Management Limited  
The copyright is held on behalf of the sponsors for the time being of the ANSA Workprogramme.





# **Tools**

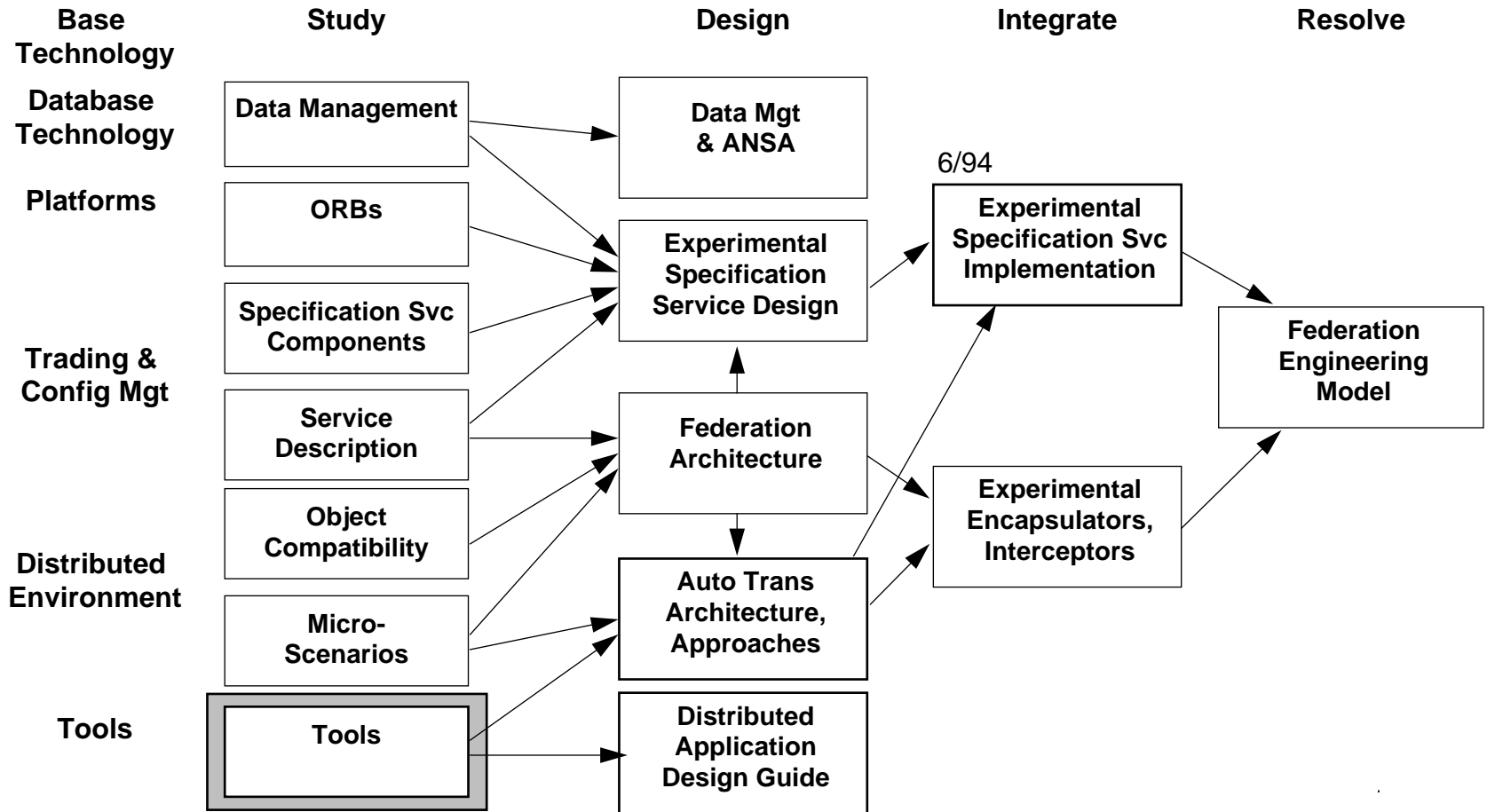
## **Work in Progress Report**

**November 2, 1993**

**Rob van der Linden**

**The Federation and Automated Transparency Topic Group**

**Accompanying paper: APM.1058**

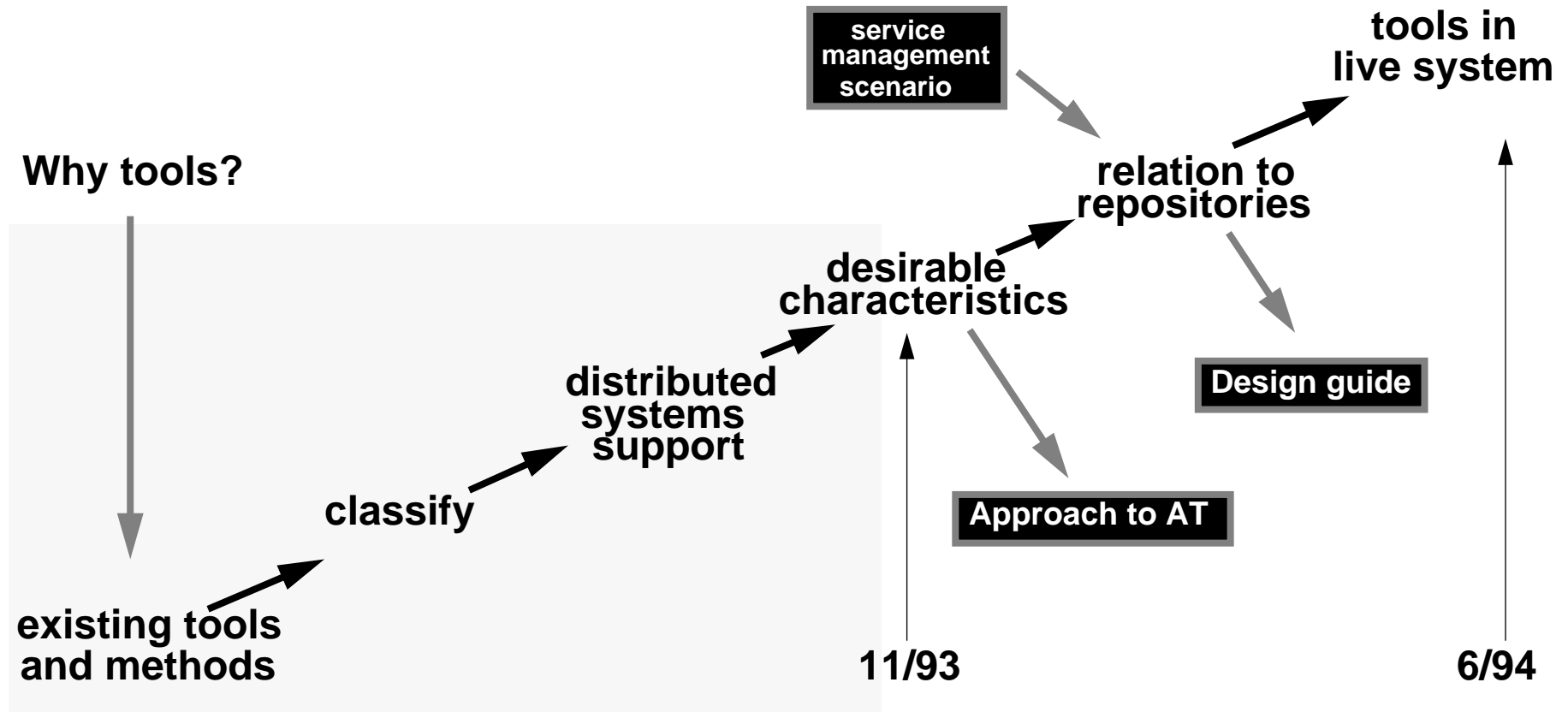




## Objective of this talk

- **Attempt to answer:**
  - can the slogan “abstract and AUTOMATE” be supported by ACCEPTABLE tools?
  - can existing tools be extended or used in novel ways?
  - can ANSA principles be applied to software tools?
  - can distributed system technology be used to improve software tools?
- **Tool users will not give up productivity aids they use now**
- **Investigate tools market and compare ideas**
- **The talk should enable you to**
  - tell us what sort of software tools are acceptable or desirable
  - understand limitations of existing tools in the distributed systems context
  - help us obtain the components for our prototype

## Detailed plan tools work



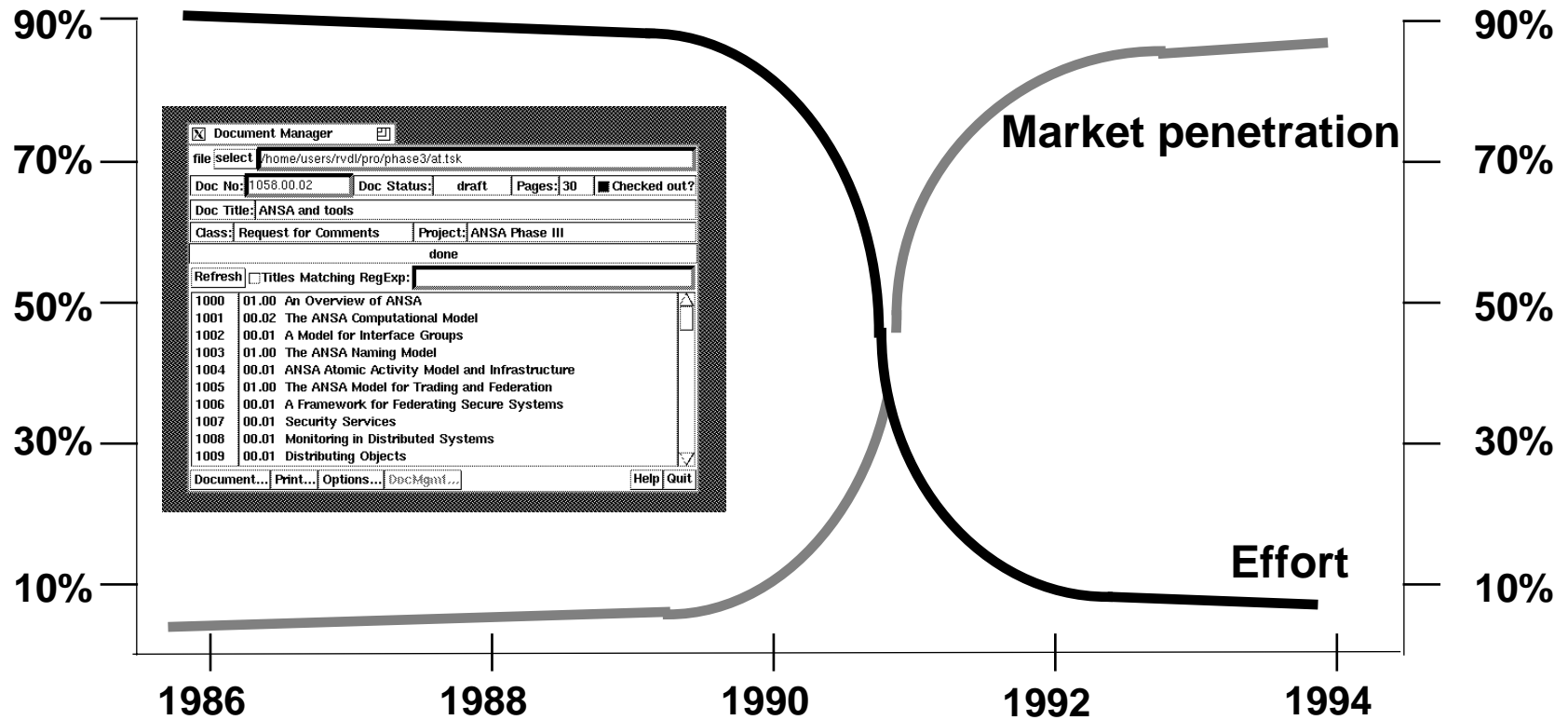


## What we looked at

- **Art\*Enterprise (from Inference Corp)**
- **Magic CAPtm (from Magic Software Enterprises)**
- **DEC/Forte (from Forte Software Inc and Digital Equipment Corp)**
- **Distributed Smalltalk (from HP)**
- **Emeraude PCTE (from Emeraude)**
- **Genera (from Symbolics)**
- **Metis (from Digital and Metis)**
- **ObjectTeam (from CADRE)**
- **OMG IDL CFE 1.2 (from Sun)**
- **ONTOS (from ONTOS Inc)**
- **OOA/RD (from Kennedy Carter)**
- **Orbix (from IONA)**
- **Ptech (from Associated Design Technology)**
- **SNAP (from Template Software & Instrumatic UK)**
- **SNiFF+ (from PtS)**
- **Statemate (from i-Logix)**
- **System Architect (from Real Time Techniques and Methods)**
- **Tenet (from Tenet Systems)**
- **VisualWorks (from ParcPlace)**



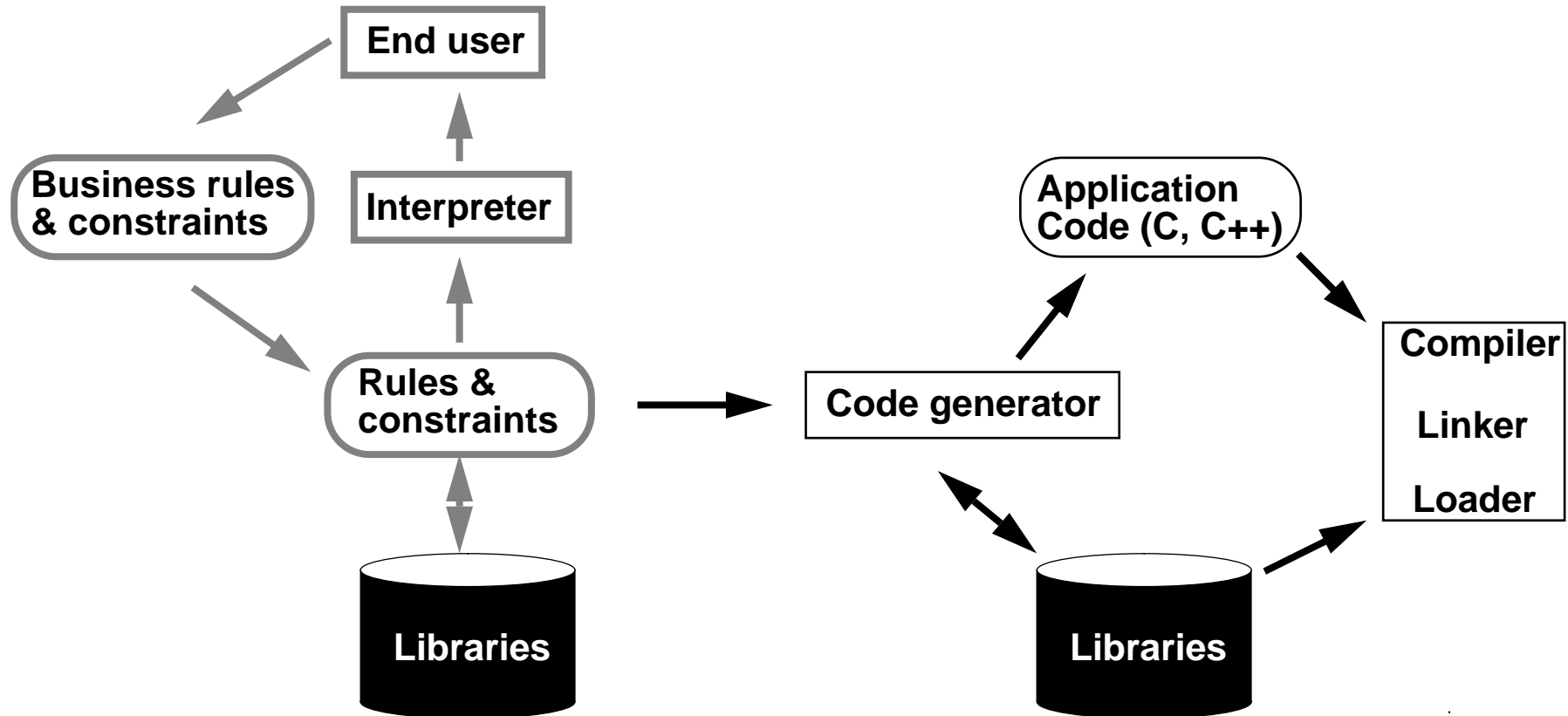
## Trends in automation: GUI programming







# Tool environment

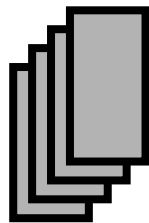


## Tool structure trend

application code

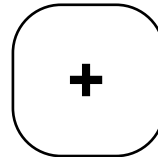


code libraries

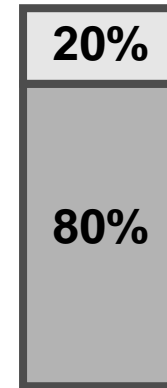


systems code

new



- construct rather than write
- configure at deployment



design time

link time

run time

epochs



# Classification

**functionality**

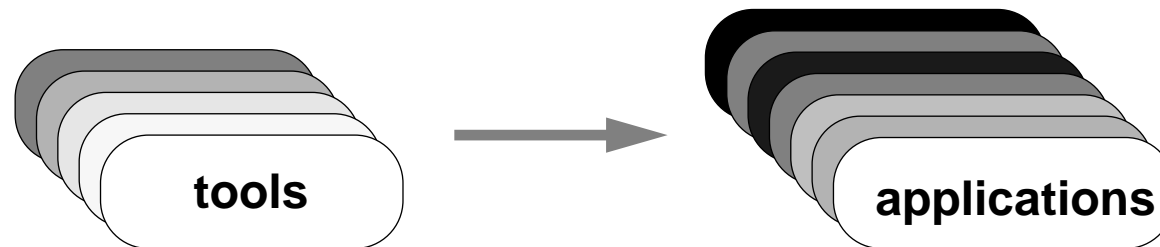
**vertical market**

**implementation**

**environment**

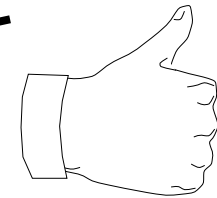
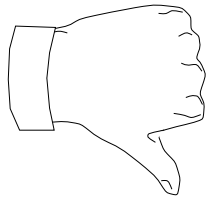
**methodology**

**object orientation**



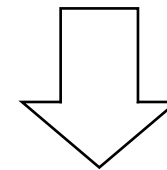
**expert systems**

## State of the Art



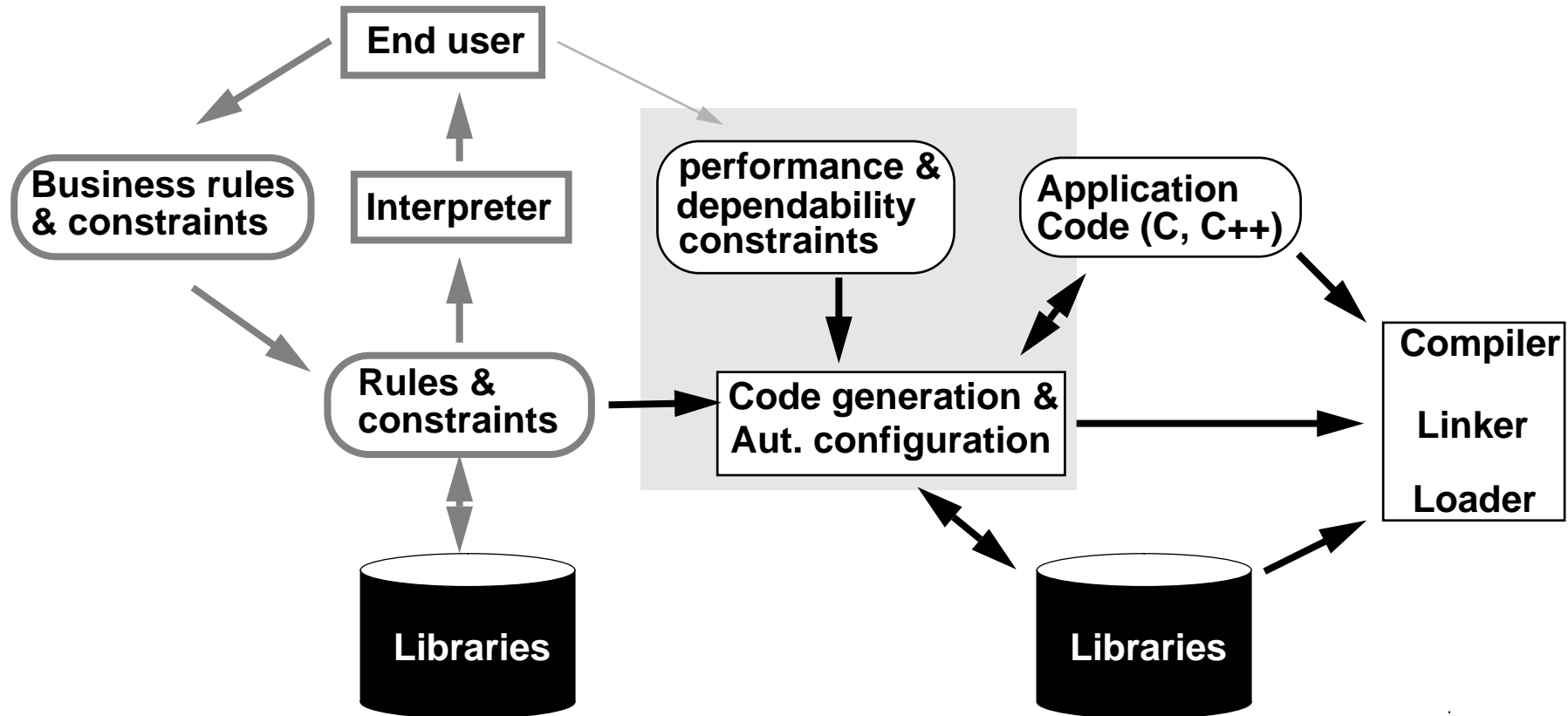
- tools limit size of application
- comms libraries out-dated (DCE support emerging)
- no agreed way to describe interfaces
- no support for application properties
- browsing limited
- tools are closed applications

- some complexities transparent
- late binding and early type checking

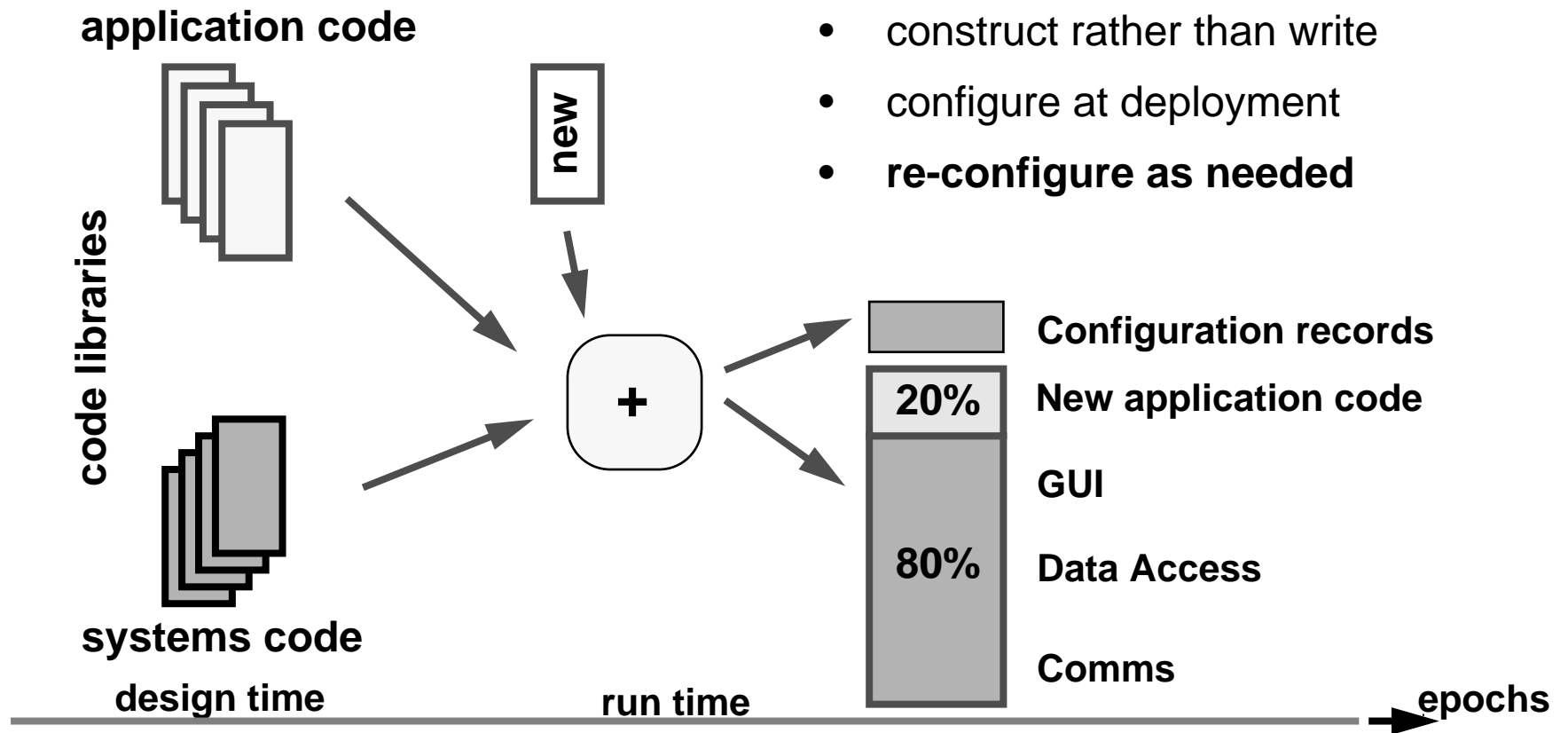


**Challenge**

# Advanced tool environment



## Advanced tool structures



# Interoperability issues

distributing the design  
& development process

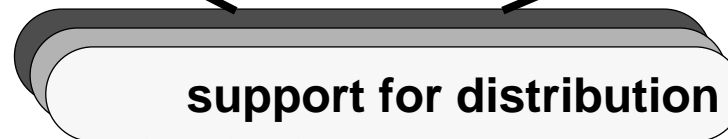
extending the design process to all epochs



## tool interworking

- specification exchange
- common libraries
- common class hierarchies

## application interworking



- OSF/DCE
- CORBA .....
- TCP/IP .....



## Summary

- **what we looked at, and what we found**
  - trend is towards automation
  - distribution (in all forms) still largely absent
- **what we want: distributed specification service**
  - total service management: extension of design & development into run-time
  - interoperability amongst tools, applications and between tools and applications
  - cooperate with programming abstraction work in other groups
  - provide framework for application properties and transformation
- **how to get it**
  - extend the trends which are already in evidence: use advanced tools
  - build on existing products (database, expert system, tooling environment)
- **first prototype mid 1994 ?**
  - what products should we use and what can we obtain from you?