Philips Software Conference

From

OO-Predevelopment

To

Product

To

Re-use

By

Gerrit Muller

Philips Medical Systems

Common Digital Systems

Predevelopment

Start: 1987/1988

Purpose: Deliver “Common viewing” components to all PMS product groups

Language: Objective-C

HW: Standard Sun workstations

Method: Prototype -> Evaluate -> Redesign -> Prototype -> etc.

+ Continuous integration

+ No formal analysis or design tools

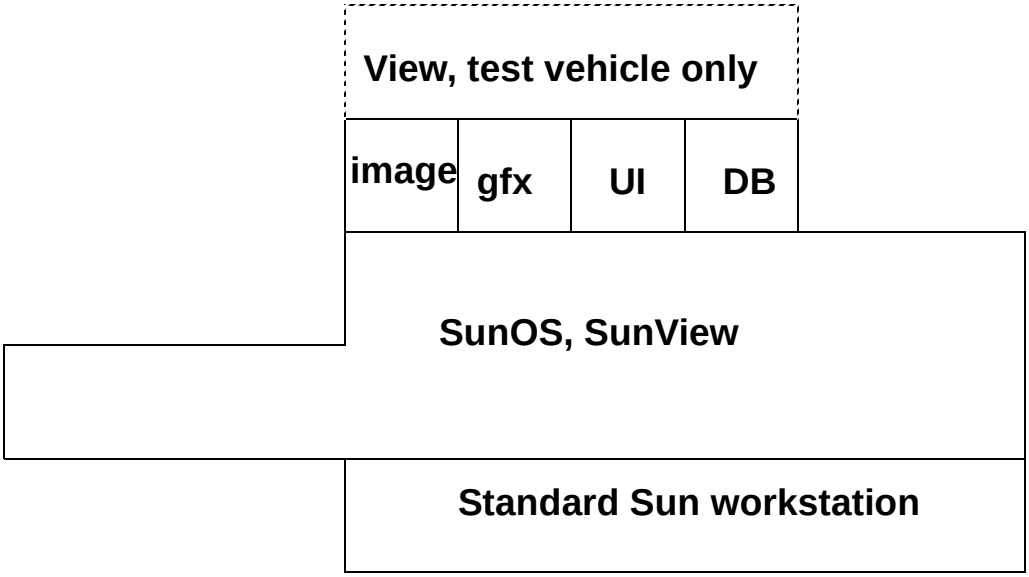
Predevelopment results:

- Powerful, flexible viewing, due to OO support for
  - + handling a mix of image types in a natural way.
  - + many parallel user activities, with own contexts
- Feasibility of using standard workstation hardware
- First learning step on OO learning curve:
  - + balance of generalization and specialization
  - + inheritance depth
  - + power and pitfalls of call-back scheduling
  - + required skills and education level
- Proto-application 1990
- Proto-product 1991

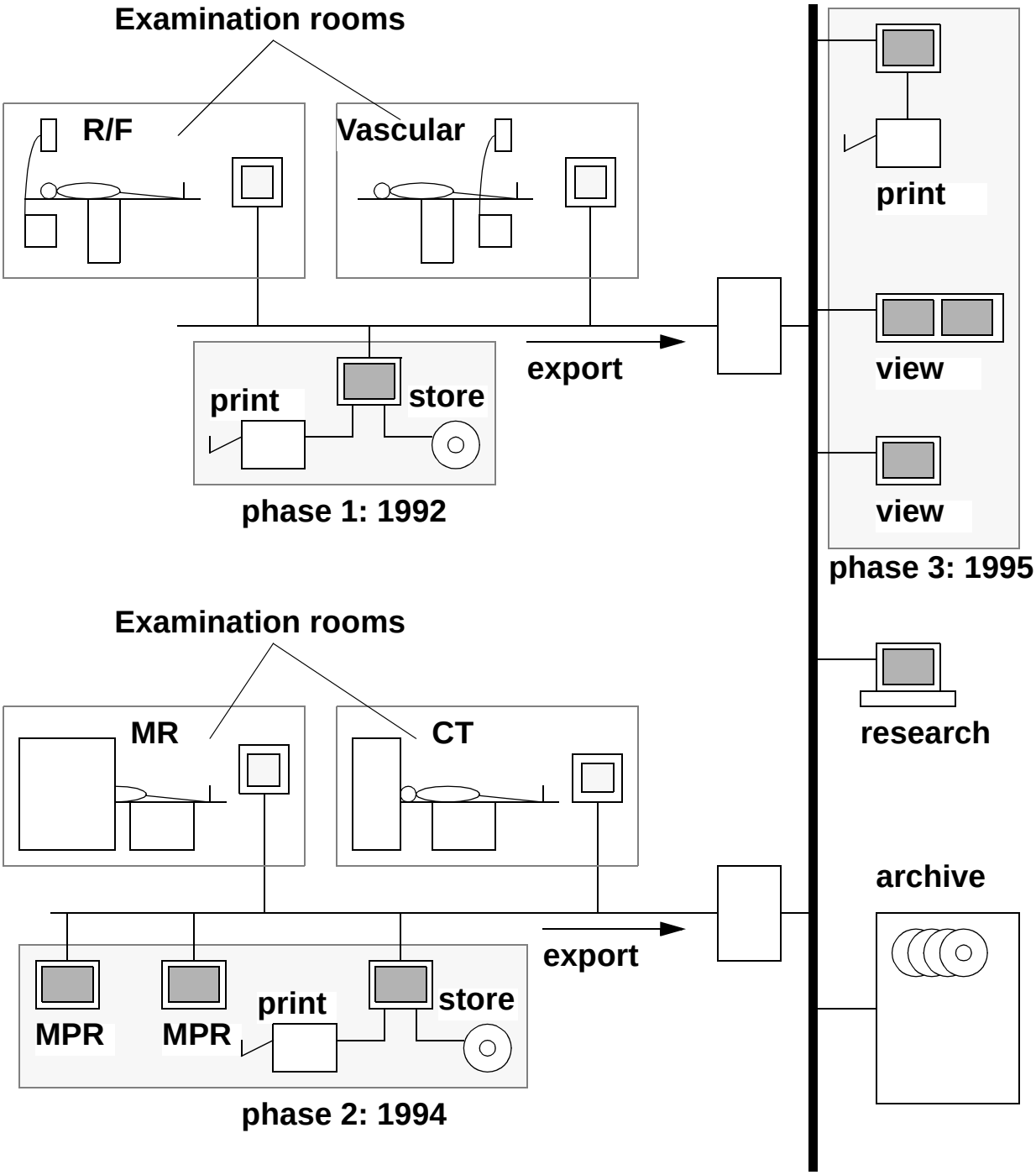
Remaining issues after pre-development

- Re-use model, process and procedures
  - + re-use by “sub-classing” of “toolbox-classes”, rewriting “application modules” and addition of product specific modules
  - + interface scope becomes very wide; A new common viewing release requires large re-do.
- Common Non-viewing functionality
- Large set of “data” files
- Several One-shot re-use projects based on proto’s. (Copy/Paste/Modify re-usability)
- Management (mis-)perception: product is finished
- Too many dependencies/cross references between modules

september 1991



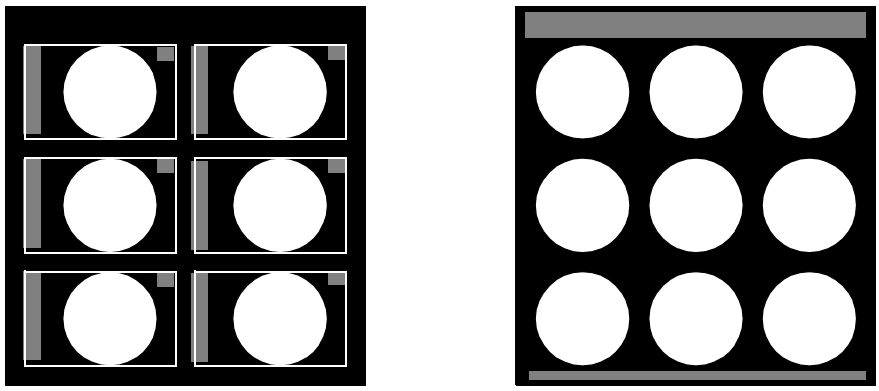
EasyVision family of products



EasyVision R/F R1

- From SW to System
- (Large) Reduction of viewing functionality
- Selling feature: Effective Film usage:

OLD NEW

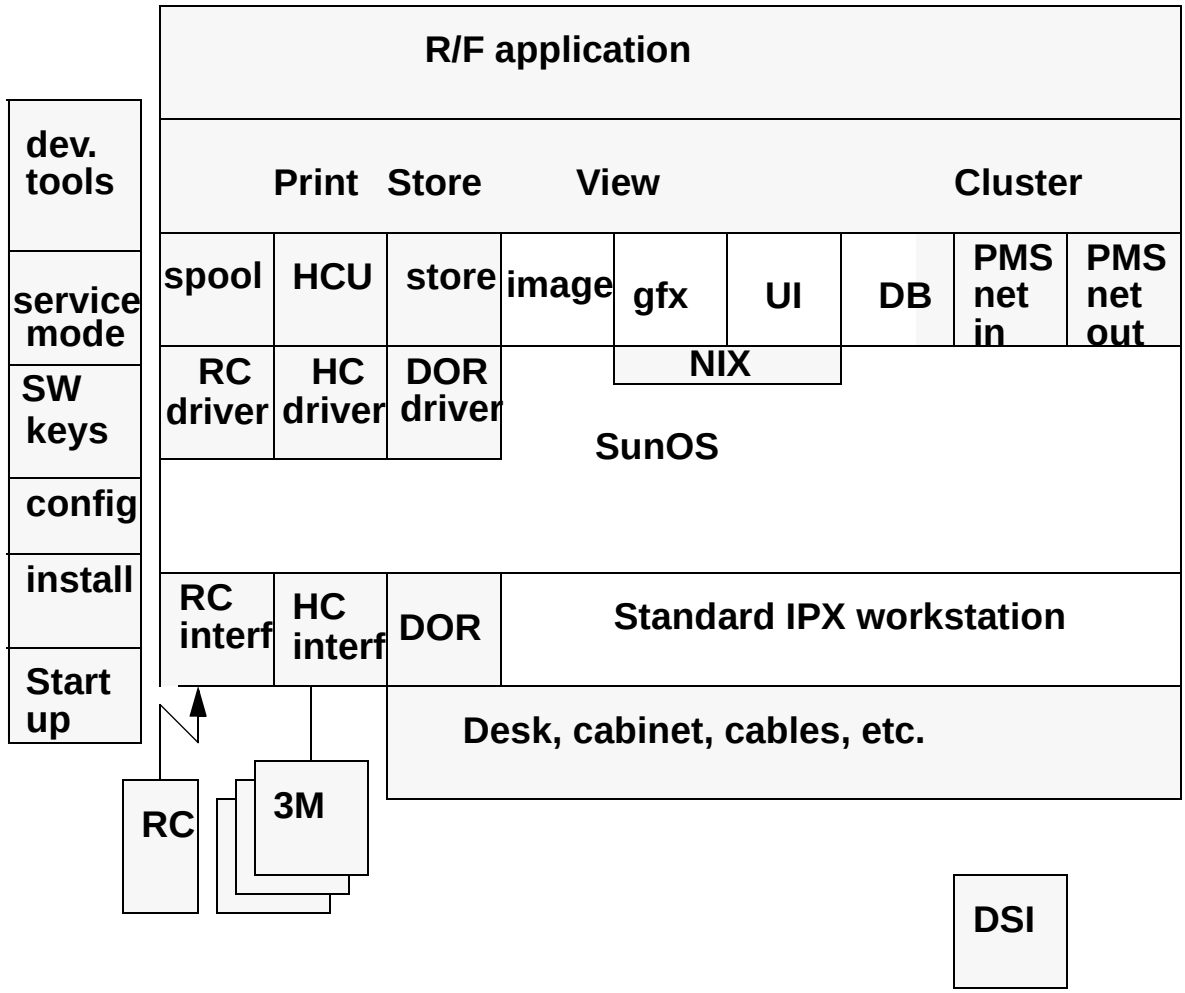


- Remote control
- Tuned for routine use (for example FIFO database, report driven printing)
- “Properties” instead of data files

Remaining issues after first product

- Re-use model
- interface management
- property management
- data model in relation with “outside” world
- modularity, dependencies

september 1992



Why workstation product group?

- Integration of functionality in hospital
  - + Multi modality
  - + Multi vendor
  - + Multi application
  - + Distributed application
- Balance of functionality inside and outside examination room
- Distributed applications, networking:
  - + scalibility
  - + graceful degradation
  - + customization

Re-use model:

A **re-usable platform** is developed and extended:

Product = tuned platform + specific SW

- Platform consolidation of application knowledge
- Re-use in look-alike products
- Exploitation of technological and organizational experience

Re-use requires:

solution of all open issues.

Technology improvement plan:

- phased improvement
- 5 manyear/year
- “External” re-use 1995 and later

Major phases

- Phase 1 (modularization):
  - + Cleanup most obvious modules
  - + First division in separate packages
  - + Equalization of internal data model and PMS Data Dictionary
  - + PMSnet, PMSdor, complete new
  - + analysis (modularity, notifications, properties)
- Phase 2 (Interfacing):
  - + Further modularity restructuring
  - + Prototyping interface
  - + Advanced development interface
  - + Prepare external interface
  - + Explore real time extensions(e.g. Threads)

Major phases 2

- Phase 3 (Internal benefit, standardization)
  - + Explore C++
  - + Explore X
  - + Implement 1D viewing
  - + Use external interface
- Phase 4 (external benefit)
  - + Decide on C++, X use
  - + Use platform by non CDS clients

Status june 1993

- Modularity
  - \+ CDS pack independent of rest SW
  - \+ SW archive divided in “groups”, dependencies are analyzed and reduced
- Property management
  - \+ file structure streamlined
- SPI support library
  - \+ Implementation finished
  - \+ Increased performance and functionality
  - \+ Much less code
  - \+ Configuration simpler
- PMSdor, PMSnet redesign and coding planned
- Solaris 2: masterplan
- HP: viewing ported, plan for product porting

- Cardio graphics:
  - + additional functionality
  - + “cold” graphics removed
- Data representation:
  - + XDR based self describing object format
- Data base:
  - + improved performance
  - + support for spooled services
- Process structure:
  - + import and export servers-> network server
  - + spoolers and UNIX command server removed
- Memory usage:
  - + ASW: 20% reduction (UNIX 20% increase)
-

What is the platform?

- All SW and HW shared by multiple products
- Managed centrally by CDS
- Own lifecycle
- Continuous change and expansion:
  - + External changes (workstations, disks, optical media, framebuffers, keyboards, system software)
  - + New functionality;

    Move from specific to re-usable
  - + Internal rework (PR’s, CR’s)

Scope of the platform

- Functional:
  - + Image and graphics

    processing, analysis, display, manipulation
  - + Films and monitors

    device handling, layout and presentation management, image quality
  - + Patient and image data

    storage, communication, data management
  - + Distributed application

    client-server architecture, system clusters, PMSnet, customization

Scope of the platform 2

- Support:
  - + Hardware

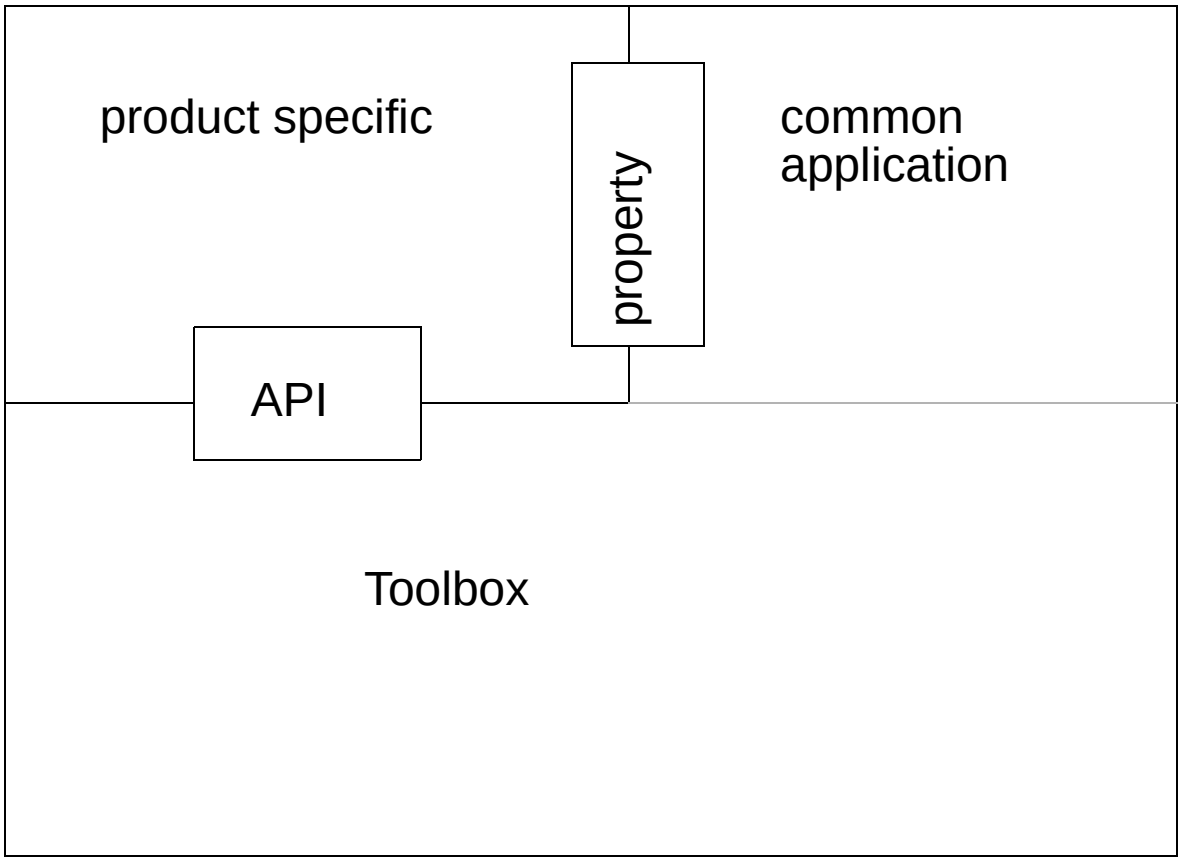
    workstation, optical media, hardcopy units incl. interface, standard video in and out, remote control, desk, spinning wheels...
  - + System software

    operating system, drivers, networking, system start up and shutdown
  - + Installation, configuration and service diagnostic SW
  - + Elementary software support

    collections, strings, symboltables, notifications, properties, error handling, etc.
  - + Libraries

    user interface, screen management, printing, database, PMSdor and PMSnet formatting and protocols, graphics, image processing, etc.

How is the platform used?

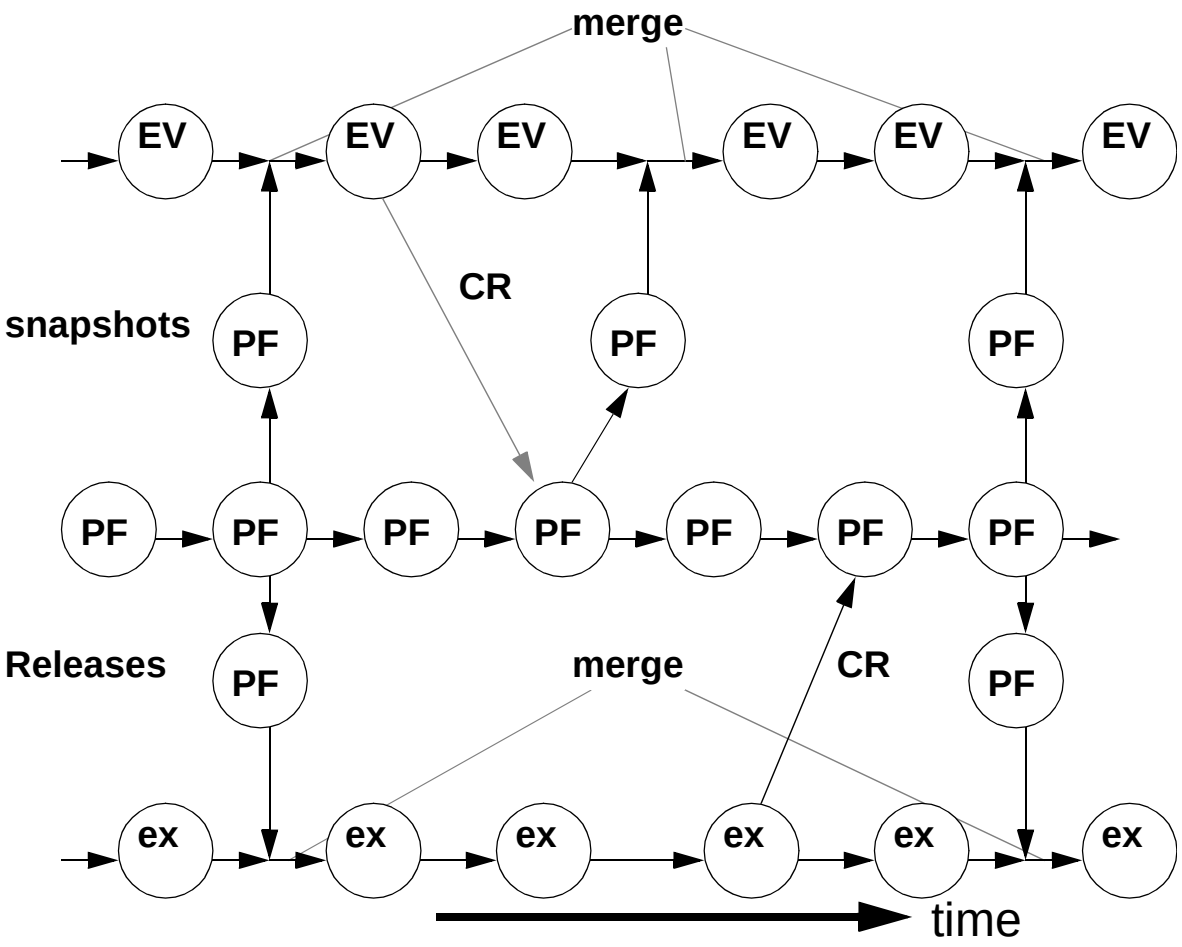


- Property interface

  \+ configure common application

- API (Application Programmers Interface)

Lifecycle and archive structure.



- Intern CDS-Best: snapshots (feedback)
- non CDS products: releases (own lifecycle)
- Old interfaces: allow in principle for transition period of 1 year (after phase 3: introduction of external interface)

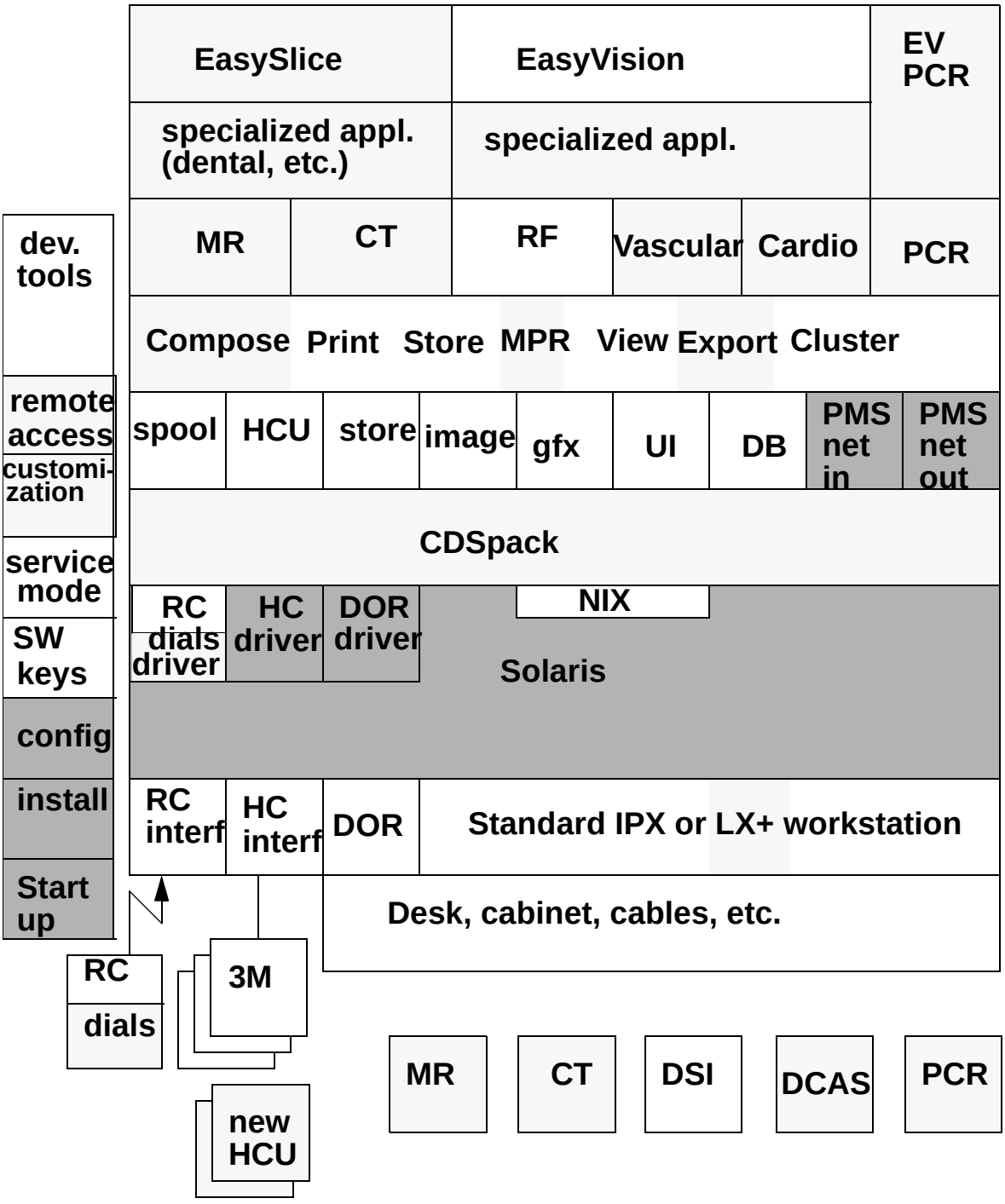
Product schedule 1994:

- Easyvision CT/MT april 1994
- Easyvision R/F, vascular, cardio june 1994
- Both products based on same platform

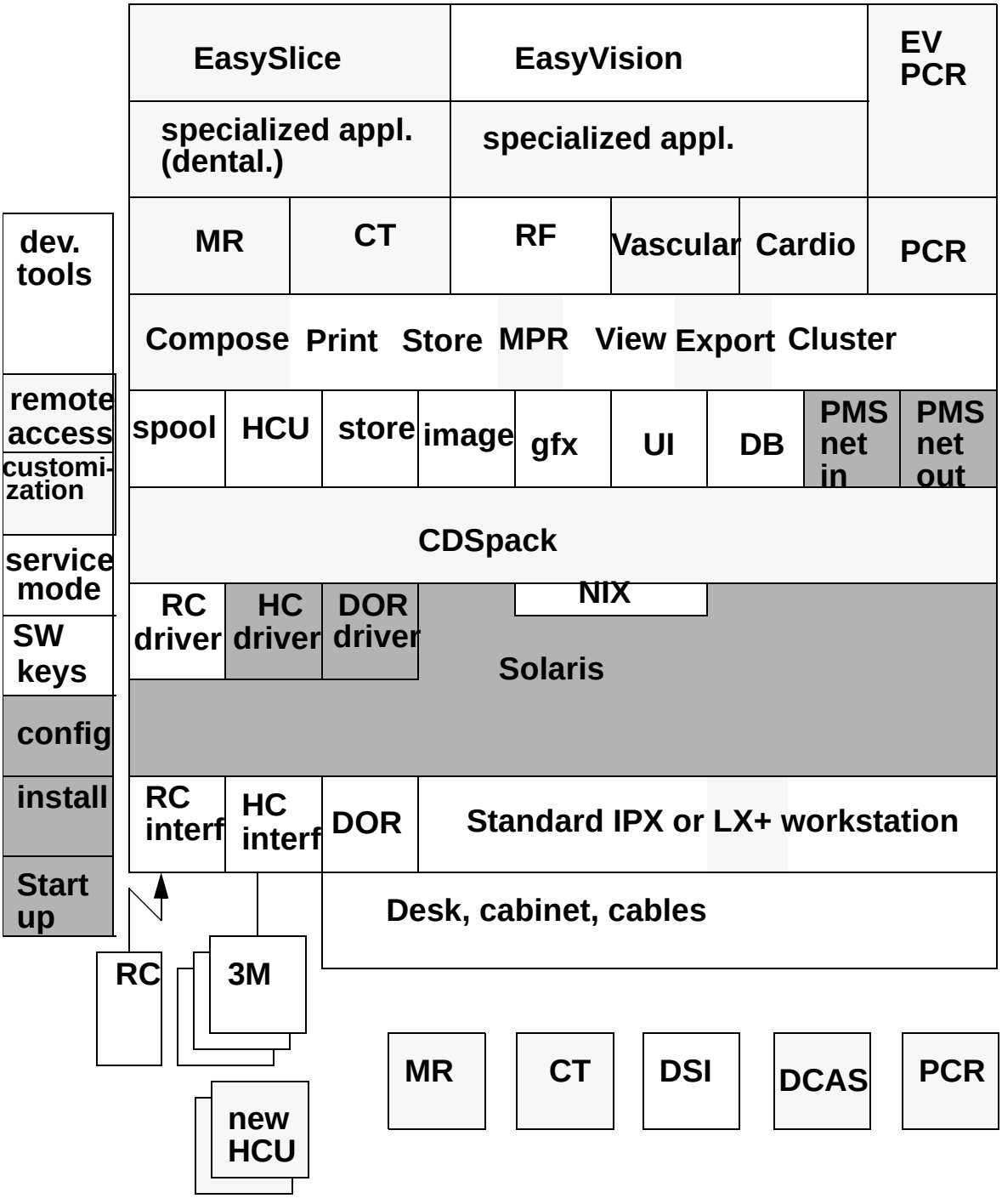
Open issues currently:

- Concurrent development of 3 archives: How to keep differences manageable, changes synchronized etc
- Re-use interface level, interface management
- management of non-C code (properties, report definitions, etc.)

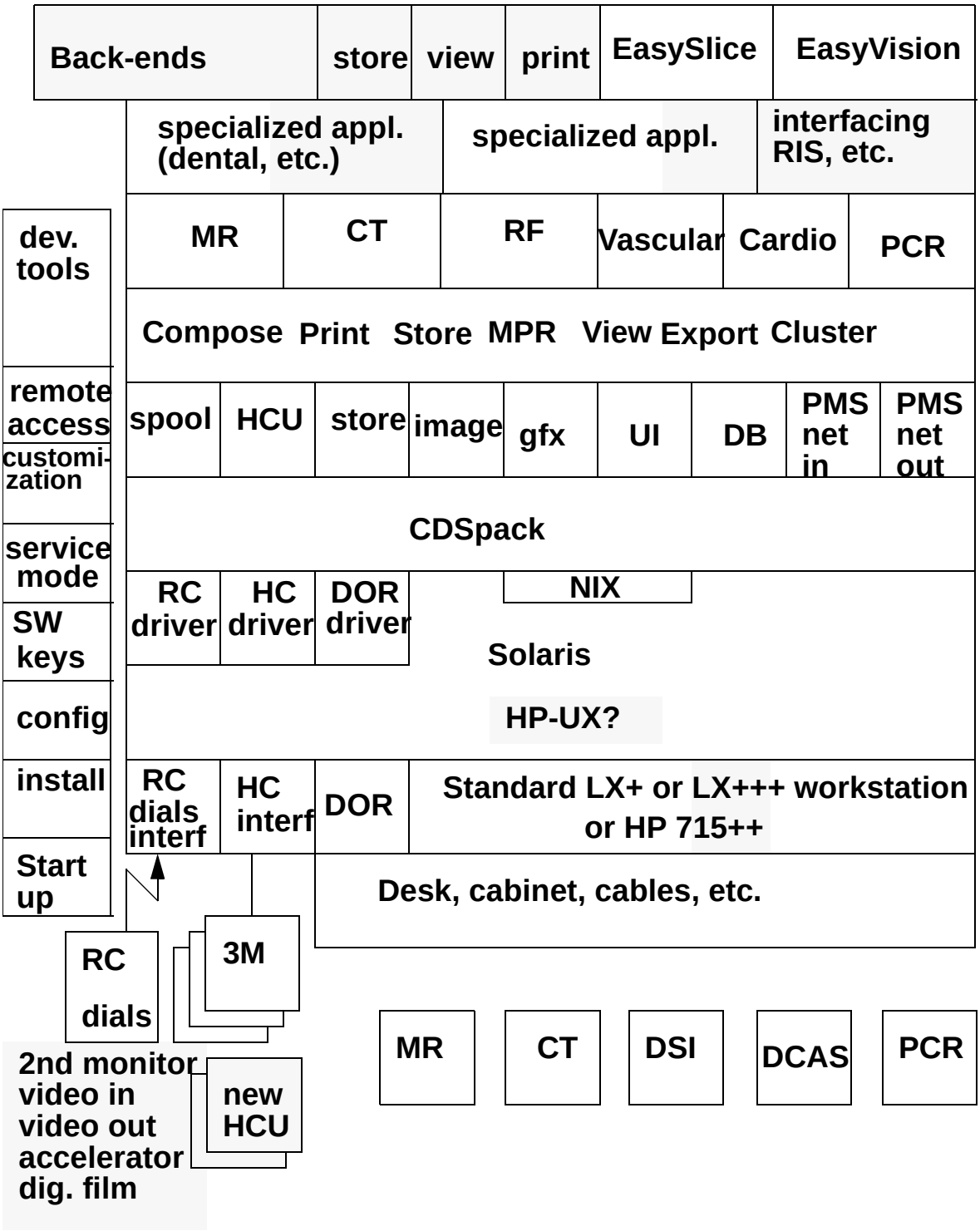
june 1994


EasySlice
EasyVision
EV PCR
specialized appl. (dental, etc.)
specialized appl.
dev. tools
MR
CT
RF
Vascular
Cardio
PCR
Compose
Print
Store
MPR
View
Export
Cluster
remote access
customi-zation
spool
HCU
store
image
gfx
UI
DB
PMS net in
PMS net out
CDSpack
service mode
SW keys
RC dials driver
HC driver
DOR driver
NIX
Solaris
config
install
RC interf
HC interf
DOR
Standard IPX or LX+ workstation
Start up
Desk, cabinet, cables, etc.
RC dials
3M
new HCU
MR
CT
DSI
DCAS
PCR

june 1994


EasySlice
EasyVision
EV PCR
specialized appl. (dental.)
specialized appl.
dev. tools
MR
CT
RF
Vascular
Cardio
PCR
Compose
Print
Store
MPR
View
Export
Cluster
remote access
customi-zation
spool
HCU
store
image
gfx
UI
DB
PMS net in
PMS net out
CDSpack
service mode
SW keys
RC driver
HC driver
DOR driver
NIX
Solaris
config
install
RC interf
HC interf
DOR
Standard IPX or LX+ workstation
Start up
Desk, cabinet, cables
RC
3M
new HCU
MR
CT
DSI
DCAS
PCR

1995/1996


Back-ends
store
view
print
EasySlice
EasyVision
specialized appl. (dental, etc.)
specialized appl.
interfacing RIS, etc.
dev. tools
MR
CT
RF
Vascular
Cardio
PCR
Compose
Print
Store
MPR
View
Export
Cluster
remote access
customi-zation
service mode
SW keys
config
install
Start up
spool
HCU
store
image
gfx
UI
DB
PMS net in
PMS net out
CDSpack
RC driver
HC driver
DOR driver
NIX
Solaris
HP-UX?
RC dials interf
HC interf
DOR
Standard LX+ or LX+++ workstation or HP 715++
Desk, cabinet, cables, etc.
RC dials
3M
2nd monitor
video in
video out
accelerator
dig. film
new HCU
MR
CT
DSI
DCAS
PCR