



# Blaise NG

First glimpse



# Key issues in current system

---

- Language enhancements
- Layout
- Unicode



# Consequences

---

- New parser and IDE
- New meta files
- New runtime system (DEP)



# Research

---

- Platforms
- IDE / Parser



# Development platform

---

- The platform that the Blaise developers use
- Choice
  - .NET



# Design platform

---

- The operating system where the Blaise development system will run upon
- Choice
  - Windows, using .NET framework 3.5



# Runtime platform

---

- The operating system where Blaise instruments run upon
- Choices
  - Windows, with .NET 3.5
  - Browsers, with Silverlight 2.0



# IDE/Parser

---

- Microsoft Visual Studio Extensibility (VSX)
- Actipro Components
- DSL tools
- Gold Parser
- ANTLR



# Phase 1

---

- Scope
  - Datamodels
  - New control centre
  - New source files
  - New parser
  - New meta files
  - Layout
  - Compatibility



# Phase 1: Datamodels

---

- Focus on datamodels
  - No Manipula/Maniplus
  - No CATI
  - No additional tools (Bascula, Delta)



# Phase 1: New Control Centre

---

- Using Actipro components
  - Flexible
  - Reinventing the wheel ☹️
  - Integration with new parser
  - Source Code Editor
  - Graphical Datamodel Designer



# Phase 1: New source files

---

- New format (.blax)
  - SQLite database
  - Can contain main source file as well as include files
  - Can contain additional information



# Phase 1: New parser

---

- Using ANTLR
  - Define grammar of the Blaise language
    - Not very easy
  - Some small changes to the language (no layout statements anymore)



# Phase 1: New meta files

---

- New format (.bmix)
  - SQLite database
  - Extensible



# Phase 1: Layout

---

- WYSIWYG approach
  - Powerpoint interface

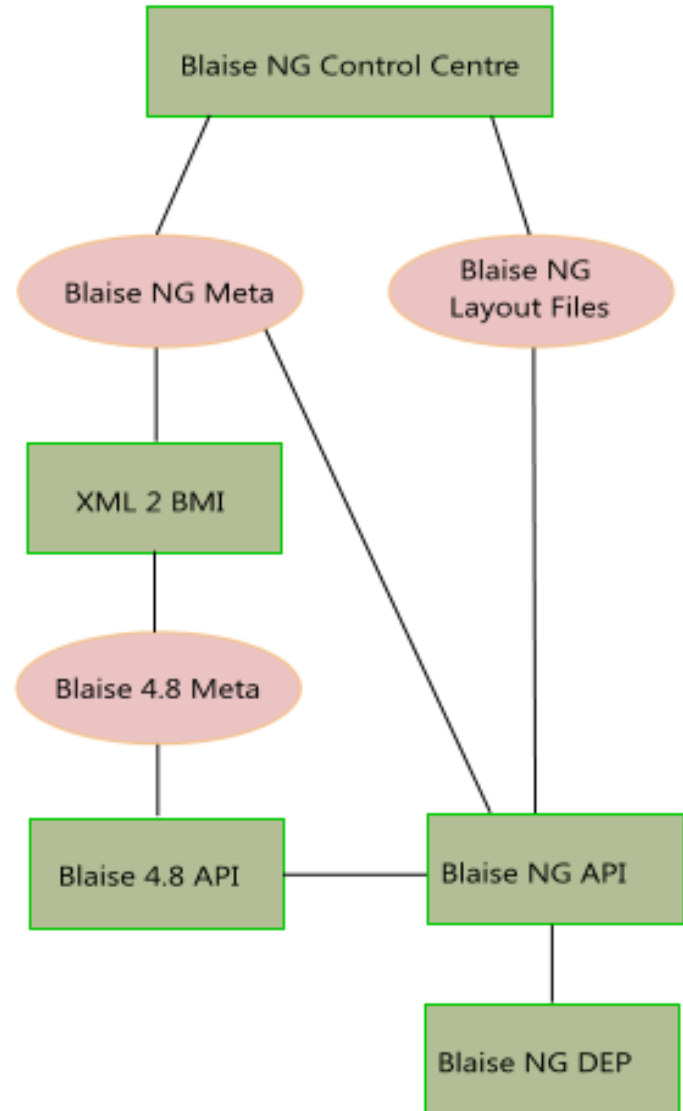


# Compatibility

---

- Language
  - Existing 4.x datamodels can be used
- Behaviour
  - Behaviour is the same as in 4.8
- Data
  - Data is interchangeable with 4.8
- Idea:
  - Reuse of the Blaise 4.8 API Service

# Architecture





Demo

---



# Phase 1: ToDo

---

- Stabilize current product
- Import tool
- Template designers
- Documentation
- Installation
- Test with real datamodels
- Incorporate user feedback



# Phase 2 and beyond

---

- Rewrite Rules Engine
- Data storage
- Language enhancements
- Localization
- Manipula
- Event language
- Multimode support / Case Management
- Debugger
- Paradata



# Milestones

---

- End of this year: CTP (result of Phase 1)
- Next IBUC: First production version