Introducing BOB the Builder

Open Source, modern, dependency-based build in the IFS of class IBM i applications

Edmund Reinhardt

Edmund.Reinhardt@ca.ibm.com

IBM i Application Development Tooling Architect

© Copyright IBM Corporation 2022

Challenges for building programs on the IBM i

- SRC-PF
  - Fixed record length
  - Not accessible to open ecosystem, including git and make
  - 10 char names
  - All source of same type stored in QxxxSRC to avoid name conflicts (member type does not disambiguate)

- Libraries
  - Only 2 level hierarchy to organize, with only short 10 char names

- Source control
  - None – sequence number dates
  - Home grown
  - Proprietary IBM i systems
    - Cost
    - Smaller market = less investment

- Build system
  - Individual CRTXXXMOD + CRTPGM
  - CL Scripts
  - A couple of vendors have dependency-based build

© Copyright IBM Corporation 2022

RDi i Projects - lessons learned

- Supports git but …

- Mapping from i Project to exactly one library was too inflexible
  - Some customers target many libraries from one project (program / data / source)
  - Other customers have huge libraries
- Metadata was very hard to maintain
  - Having a parallel directory hierarchy under .ibmi meant that any time a SRC-PF or member changed names the metadata was lost
- Mapping rigidly to SRC-PF meant inheriting all of its limitations
  - Fixed line length
  - Fixed directory hierarchy of basically 1 level with only 10 characters
- Build was very limited,
  - No disambiguating of PGM vs MOD
  - No understanding of binding relationships
  - No incremental ability – i.e. only build what had changed

© Copyright IBM Corporation 2022

Existing bob by S4i

- Fascinating open-source project by Jeff Berman then of S4i https://github.com/s4isystems/Bob
  - Incremental compile ability based on gmake
  - Some level of ILE binding understanding
  - No longer bound to library and SRC-PF structure and naming limitations
  - Member level specific metadata using gmake variable
  - Consideration of target EBCDIC CCSID for compiler
  - Support of old languages whose compilers do not have IFS support yet (DDS, UIM)
  - Retrieval of all EVFEVENT files to enable compiler feedback

- Limitations
  - Uppercase names required
  - Single target library
  - Single directory containing source
  - No metadata on environment prerequisites,
    i.e. LIBL, where to find includes, ASP etc.
  - Install was complex – not yum-enabled
  - No 1 to 1 mapping of file extensions to compile
    (i.e. are we targeting MOD or PGM)

© Copyright IBM Corporation 2022

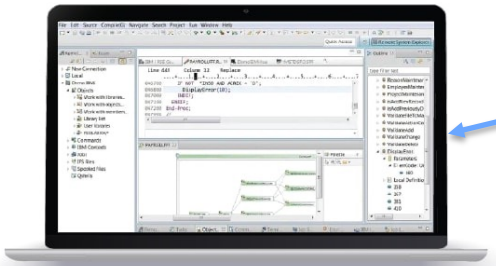
Enhancement

- Project definition
  - Know how to build yourself
  - Know where to resolve includes
  - Know how to set up environment
  - Learn from package.json
  - But make it flexible so that what is stored in git does not have to be modified from 1 developer or deployment scenario to another
- No limit on number of directories and their nesting
- No limit on directory naming
- No limit on number of object libraries
- Unambiguous mapping from file name to compile type

© Copyright IBM Corporation 2022

Use cases: Bob should be usable by …

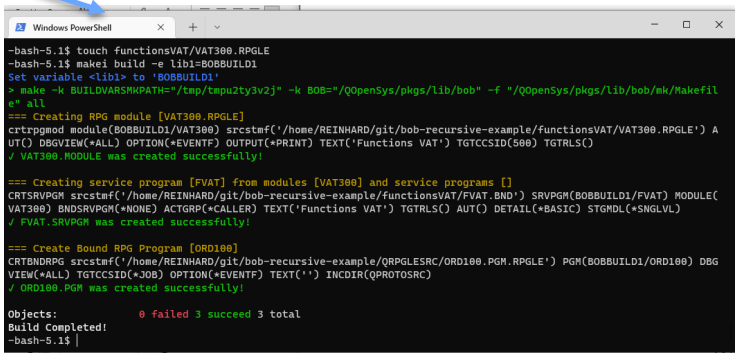
- PASE command line
- Windows/Mac command line with rsync/scp to do file transfer
- Any VS Code extension for IBM i development like Code for i (halcyon tech)
- RDi
- MERLIN!!



RDi

Visual Studio Code

Merlin IDE



© Copyright IBM Corporation 2022

© Copyright IBM Corporation 2022

Project Definition

- The vision
  - ILE were as portable as any other open source package
  - You just did a **git clone** + set some required variable to specify what is unique about your environment … and you were ready to build and deploy
- The reality – **iproj.json**

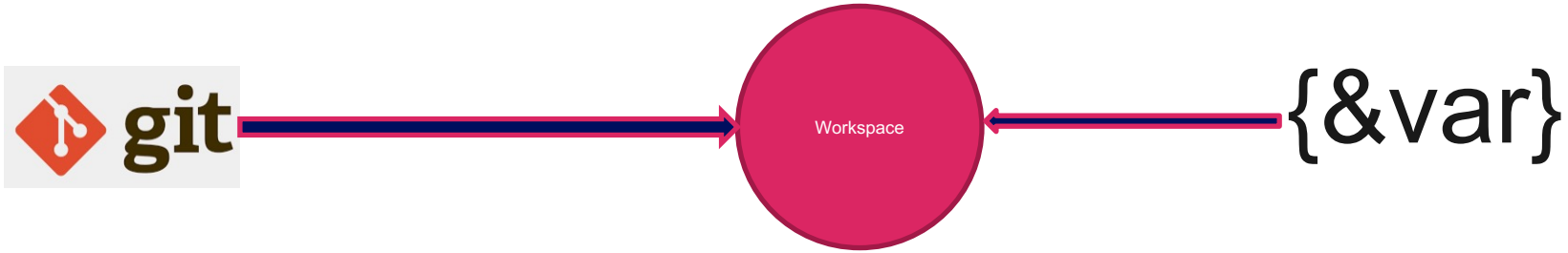
© Copyright IBM Corporation 2022

Workspace no longer a point of failure!

- In the past, if you lost your PC or developer library on the IBM i – catastrophe

© Copyright IBM Corporation 2022

Now everything is securely backed up elsewhere

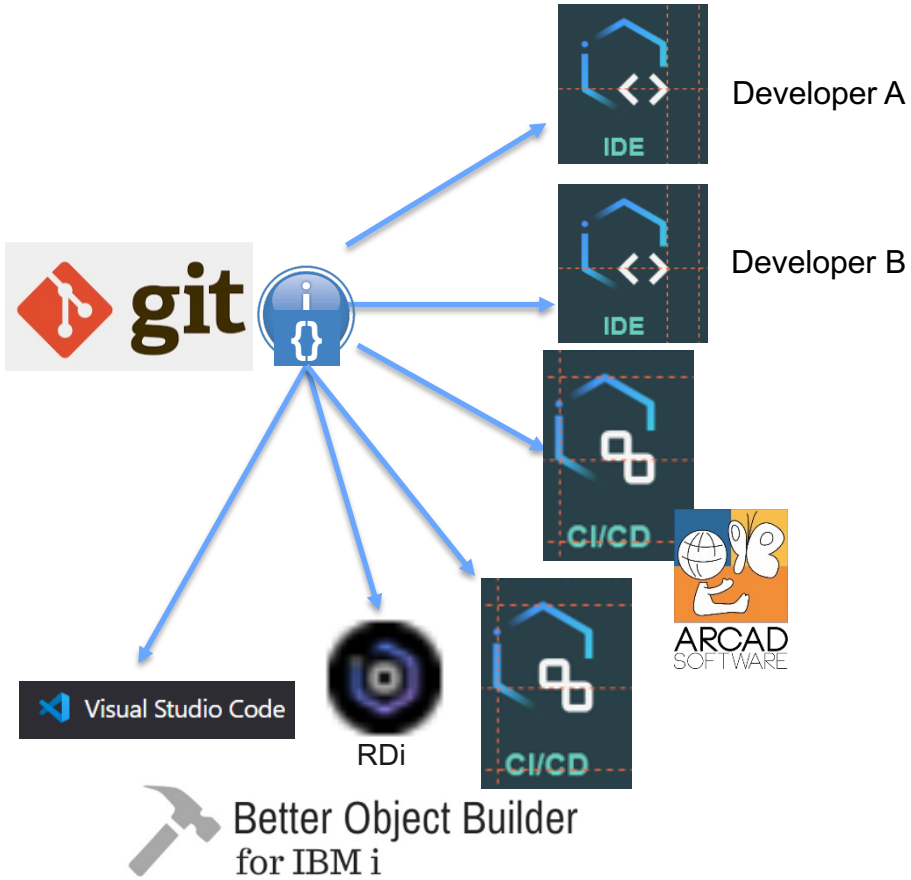


- Source
- Object definitions (CL, SQL) as source
- Build definitions (ibmi-bob) as source

- Workspace specific libraries, directories or commands

© Copyright IBM Corporation 2022

iProj – portable, git-storable project

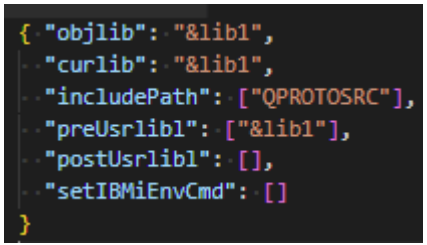


- Single project definition that can be used in many contexts
  - Cloned into different development environments
    - Merlin
    - VS Code
    - Rdi
  - Used to define build in CI/CD pipelines
    - Buildable by ARCAD builder
    - Buildable by open source BOB
  - Has to specify requirements on external environment
    - LIBL
    - Where to resolve includes
    - ASP, OVRxxx etc.
  - But has to be configurable for each environment
    - Uses &name env variables
  - Intelligent defaults to minimize metadata

© Copyright IBM Corporation 2022

Project definition specifics

- Full documentation is here https://ibm.github.io/ibmi-bob/#/prepare-the-project/project-metadata



- Some attributes
  - **objlib –** what library should objects be compiled into by default

  Setting up of build/runtime environment
  - **curlib**
  - **preUsrlibl**
  - **postUsrlibl**
  - **setIBMiEnvCmd**

  - **includePath** – what directories should tools or compilers look into for include files

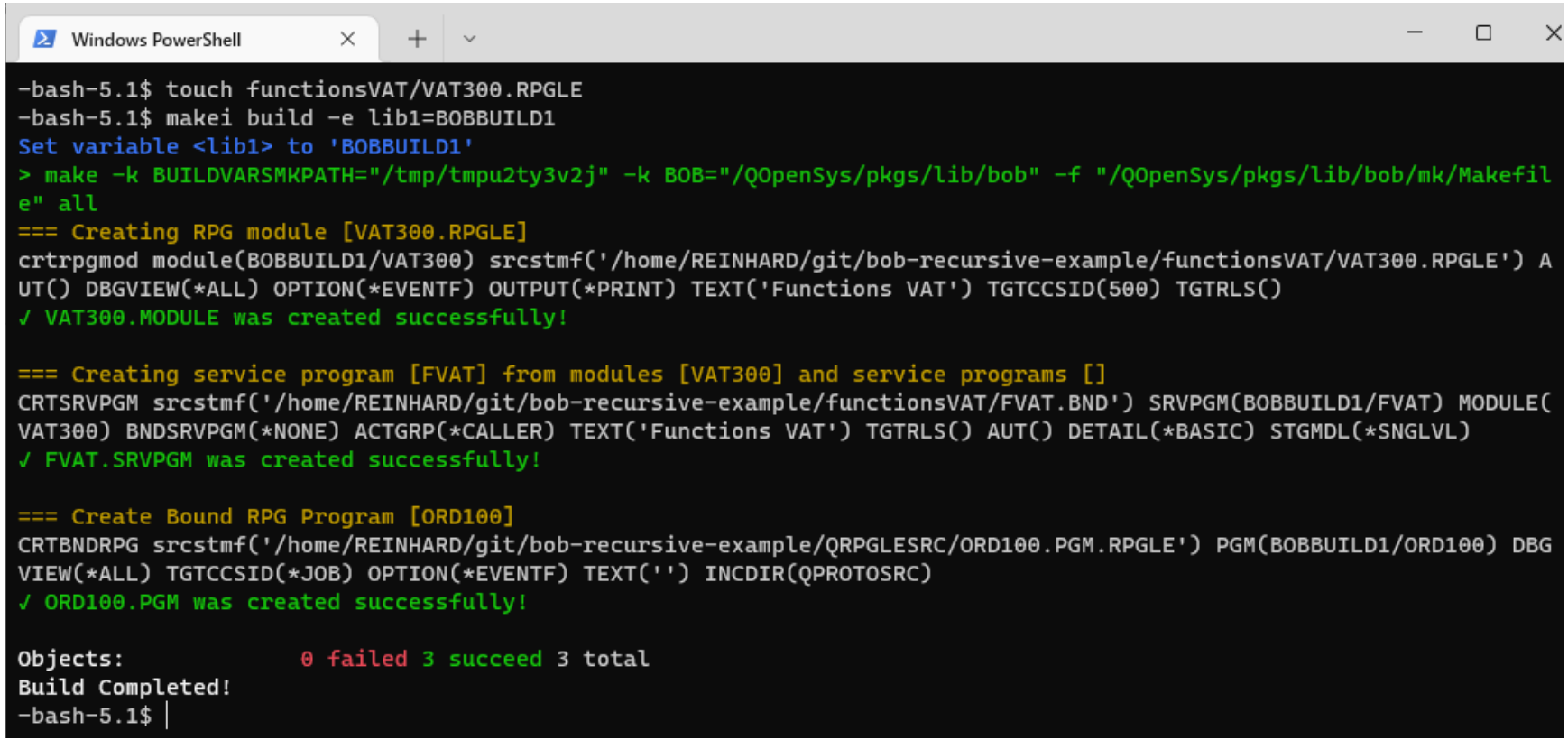
© Copyright IBM Corporation 2022

Demo project structure

- A workspace can contain multiple projects.
  - A directory containing iproj.json is a project
  - Projects cannot be nested
- Target library for creating objects is set by “objlib” attributed
- Can specify multiple target object library
  - Use .ibmi.json to specify target object library for a directory

© Copyright IBM Corporation 2022

Incremental build



© Copyright IBM Corporation 2022

Types of source supported

- https://ibm.github.io/ibmi-bob/#/welcome/features?id=supported-object-types

Object Type	File Extension
*CMD	.CMDSRC
*FILE	.DSPF, .LF, .PF, .PRTF
*MENU	.MENU
*MODULE	.C, .CLLE, .RPGLE, .SQLC, .SQLRPGLE
*PGM	.RPG
*PNLGRP	.PNLGRPSRC
*QMQRY	.SQL
*SRVPGM	.BND
*WSCST	.WSCST

Object Type	File Extension	CL Command
*MSGF	.MSGF	CRTMSGF + ADDMSGD
*BNDDIR	.BNDDIR	CRTBNDDIR
*PGM	.ILEPGM	CRTPGM
*SRVPGM	.ILESRVPGM	CRTSRVPGM

SQL Type	QSYS Object	File Extension	SQL COMMAND
TABLE	*FILE	.TABLE	CREATE OR REPLACE TABLE
VIEW	*FILE	.VIEW	CREATE OR REPLACE VIEW
PROCEDURE	*PGM	.SQLPRC	CREATE OR REPLACE PROCEDURE
FUNCTION	*SRVPGM	.SQLUDF	CREATE OR REPLACE FUNCTION
FUNCTION	*SRVPGM	.SQLUDT	CREATE OR REPLACE FUNCTION
TRIGGER	*PGM	.SQLTRG	CREATE OR REPLACE TRIGGER
ALIAS	*FILE	.SQLALIAS	CREATE OR REPLACE ALIAS
SEQUENCE	*DTAARA	.SQLSEQ	CREATE OR REPLACE SEQUENCE

© Copyright IBM Corporation 2022

Start Merlin project from scratch

makei init

```
-bash-5.1$ ls
-bash-5.1$ makei init
This utility will walk you through creating a project.
It only covers some common items.

Press ^C at any time to quit.
descriptive application name: (scratch)
git repository:
include path, separated by commas:
What library should objects be compiled into (objlib): (*CURLIB)
What EBCDIC CCSID should the source be compiled in: (*JOB)
curlib: (*CRTDFT)
Pre user libraries, separated by commas:
Post user libraries, separated by commas:
Set up commands to be executed, separated by commas:
license:

The following files will be added to the project
+ /home/REINHARD/demo/scratch/iproj.json
+ /home/REINHARD/demo/scratch/Rules.mk
Continue? (yes) yes
-bash-5.1$ ls
Rules.mk  iproj.json
```

© Copyright IBM Corporation 2022

Copy QSYS source into BOB projects in the IFS

- **makei cvtsrcpf** *src-pf-name library-name*
- https://ibm.github.io/ibmi-bob/#/prepare-the-project/convert-source-code

© Copyright IBM Corporation 2022

Multi-directory build

- Typical projects have multiple folders of source
- One folder may well depend on objects build from another
- Therefore need to have intelligent build that spans multiple directories
- We can can span multiple scopes

  - source
    - **makei compile –f *source.rpgle*** ➔ will build SOURCE.MODULE
    - **makei compile –f *mydirectory*** ➔ will build all the source in *mydirectory*
  - object
    - **makei build –t *target.pgm*** ➔ will build TARGET.PGM
    - **makei build** ➔ will build the whole project

© Copyright IBM Corporation 2022

Useful links

- BOB documentation
  - https://ibm.github.io/ibmi-bob/#/
  - Includes how to install, how to set up your source, how to integration with VS Code, etc.

- BOB github where you can submit issues, have discussions, read the source, make your own contributions, subscribe to be notified of the latest
  - https://github.com/IBM/ibmi-bob

- YouTube on getting started with BOB, hosted by Liam Allan
  - https://www.youtube.com/watch?v=OojAydEUWIU

- YouTube channel with more videos on BOB and Code for i from Liam
  - https://www.youtube.com/@liamallan12345

- Merlin documentation and sandbox – integrates with BOB
  - https://ibm.github.io/merlin-docs/

- Code for i VS Code extension that integrates with BOB
  - https://github.com/halcyon-tech/vscode-ibmi

© Copyright IBM Corporation 2022

© Copyright IBM Corporation 2022

© Copyright IBM Corporation 2022