Concept reflectometry reduction

- Requirements (until now) equal to Cosmos software
- ISIS reflectometry reduction (**new!**), useful!
- **Loader**: wavelength conversion (optional), scattering angle, direct beam (optional), x-axis correction (depends on direct beam)
- **Reduction**: unit conversion, division by direct beam, monitors, background, foreground ...)

Why loading direct beam? Cosmos code

temp1 = cosmos_anal_correctdistance(peakref, parref.x_min, parref.pixeldensity, parref.tofd)

temp2 = abs(temp1 - cosmos_anal_correctdistance(peakdir, pardir.x_min, pardir.pixeldensity, pardir.tofd))

' Difference in corrected **TOF distance between direct and reflect beams** is ' temp2

if (temp2 / temp1) gt 0.01 then ' Run no. ' runno ': Different TOF distances from direct and reflect runs.'

lambda = 1e10 * (c_params.planckperkg * ((findgen(tsize) + 0.5) * parref.channelwidth + parref.delay) / temp1)

D17

LR: LoadILLReflectometry
RRO: ReflectometryReductionOne version 2
RROA: ReflectometryReductionOneAuto version2

Lamp/Cosmos	Mantid (ISIS)	Mantid ILL
single file inspection/visualization		✓ (LR)
pixels vs wavelength	vs TOF	✓ (LR) or vs TOF
DB + RB	DB & RB (one file)	DB optional + RB (LR)
binning	via monitor range (RRO), user defined (RROA)	RRO, RROA
grouping user controlled via ROI	RRO	default 2-258 (RRO)
wavelength range	RRO	default 2-27 A (RRO)
ScatteringAngle san, dan, table & coherent, incoherent	via instrument definition	new input options (LR)

Attention: in Mantid, only x-values are subject to binning, thus detectors/pixels must be on the y-axis (difference to LAMP)

Differences ILL, ISIS

ISIS

- detector moving can be done during reduction (SpecularReflectionPositionCorrect v2)! Scattering angle can be computed during reduction - avoided by setting the ThetaIn input option in RROA
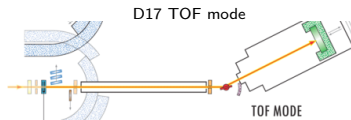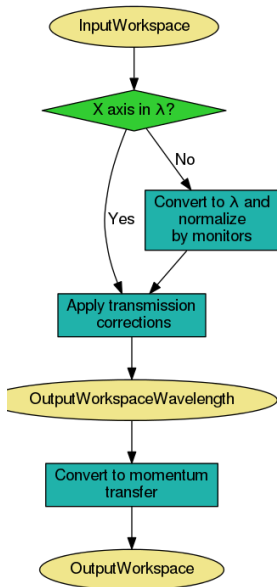- DB and RB form one workspace

ILL

- loader responsible of detector moving
- fitting used for computing the scattering angle via the detector angle

Default settings

- choose defaults according to facility and instrument
- other possibilies ...

ReflectometryReductionOne

Flowchart:

- InputWorkspace
- X axis in λ?
  - No → Convert to λ and normalize by monitors
  - Yes
- Apply transmission corrections
- OutputWorkspaceWavelength
- Convert to momentum transfer
- OutputWorkspace

D17 TOF mode

TOF MODE

- Conversion to wavelength AlignBins true
- Conversion to momentum transfer AlignBins false – Rebinning in RROA
- Division by direct beam
- Division by integrated monitor or monitors (then detectors are rebinned to monitors)

Reflectometry status (D17):

- loader: monitor errors, wavelength conversion ($+$ unit test), scattering angle ($+$ unit test), new inputs (to do: unit tests)
- reduction: first basic runs, input checks (from Nexus file: Lamp logs target_value, Mantid uses value)

Open PR (no reviewer):
fix template usage (for LoadHelper.cpp)

To do:
load Figaro data (LoadILLReflectometry)

Python script for basic reduction and comparison

```
DB = LoadILLReflectometry('/home/cs/reimund/Desktop/D17/317369.nxs')
A1 = LoadILLReflectometry('/home/cs/reimund/Desktop/D17/317370.nxs')
DB = CorrectTOFAxis(DB, ReferenceWorkspace=A1)
x = DB.readX(0)
for i in range(DB.getNumberHistograms()):
DB.setX(i, x)
DB, DB_lam = ReflectometryReductionOne(DB, WavelengthMin=2.5, WavelengthMax=27.0,
ProcessingInstructions='2-257')
DB = ConvertUnits(DB, Target=Wavelength, AlignBins=True) (Loader)
DB = CropWorkspace(DB, Xmin=2.5, XMax=27.0)
DB = GroupDetectors(DB, GroupingPattern ='2-257')
Res = Divide(LHSWorkspace=A1, RHSWorkspace=DB)
ResQ = ConvertUnits(Res, Target=MomentumTransfer)
```

Initial comparison Cosmos Mantid (**not updated**)

- Data after loading correct
- TOF x-axis with minor differences due to different computations
- Scattering angle corrected

Initial comparison of basic reduction steps (selection wavelength range, axis conversion, binning)
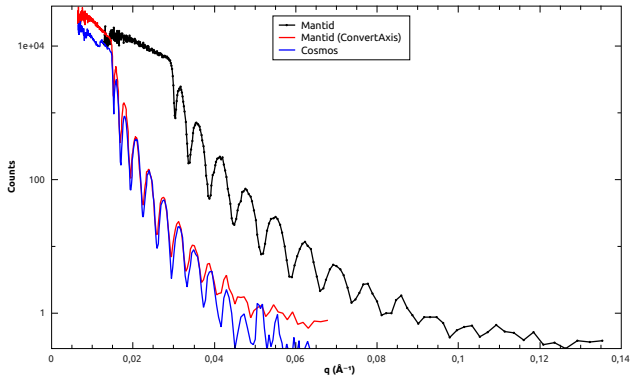
Mantid algorithms for reflectometry

ConvertToReflectometryQ (corrects detector position! ILL loader sets incident theta angle stheta)
CreateTransmissionWorkspace
FindReflectometryLines
ISIS_Reflectometry
PolarizationCorrection
ReflectometryReductionOne
SpecularReflectionCalculateTheta (output 2theta)
SpecularReflectionPositionCorrect v2 (moves detector according to 2theta)
Stitch1D
Stitch1DMany

ISIS:
CalculateResolution
CalculateSlits
CreateTransmissionWorkspaceAuto v2
ReflectometryReductionOneAuto (rebin, scale)
- move detectors if theta is given?
- if theta is not given calculate via SpecularReflectionCalculateTheta
- Line 374: theta calculated
- Line 503: CalculateResolution: TwoTheta = theta

SNS:
RefLReduction,

RefReduction
RefRoi