

Sheet1

<b>VEGAS Pulsar Mode Integration Tasks</b>		days
<b>VEGAS Manager Updates</b>		<b>50</b>
1.1	Coherent Mode backend class; update backend base class with concept of 'roach ownership'	
1.2	Incoherent Mode backend class	
1.3	Add parameters unique to pulsar modes with dep/chk/act methods	
1.4	Add samplers specific to GUPPI BOF's??	
1.5	Add parameters unique to pulsar modes in VEGAS coordinator	
1.6	Data buffer API update	
1.7	Config/Status keyword update	
1.8	Unit tests for all backend classes (analogous to DIBAS unit tests)	
<b>Configtool Updates</b>		<b>5</b>
2.1	Add pulsar specific keywords to VEGAS config class	
2.2	Identify/implement mechanism to delineate pulsar from spectral line modes	
2.3	Routing, parameter setup, checks,	
<b>Displays</b>		<b>11</b>
3.1	Bank status aggregation	3
3.2	Signal level display (e.g. Vegasdm + ?)	3
3.3	GBT status updates	5
<b>VEGAS/GUPPI HPC Cleanup/Merge</b>		<b>54</b>
4.1	Testing strategies to confirm operation (unit and system tests, test signal datafile)	3
4.2	Identify common codes & merge into library (e.g. date/time, thread management etc.)	5
4.3	Design component (stage) control/status/data API	8
4.4	Prototype ZMQ based pipeline, evaluate performance	3
4.5	Implement component API (general routines usable for any component)	10
4.6	ZMQ based keyword value service for configuration/status	3
4.7	Revise processing stages to use component API	10
4.8	Component Unit tests	7
4.9	System level tests	5

<b>Notes</b>	
[1]	By the term 'stage' meaning the processing portion of the pipeline, as opposed to the data transport mech