

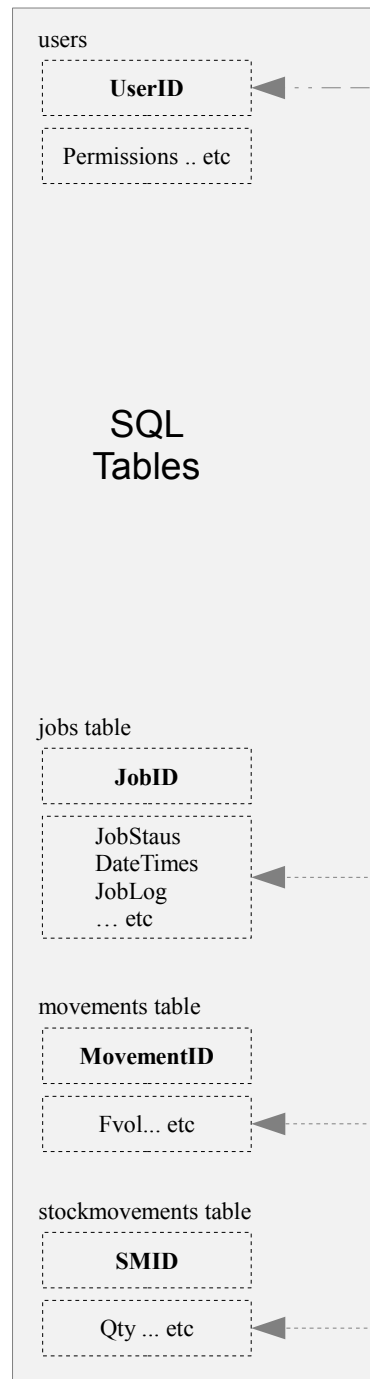
# Indevin Bulls-i Mobile System

Supervisor Console can be included in the Mobile Service Program or as part of the Users MRP System.

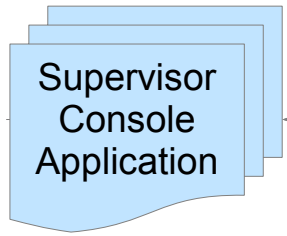
Here its the latter ie built in to Bulls-i Desktop Program

Controls which user is assigned to device and their permissions.

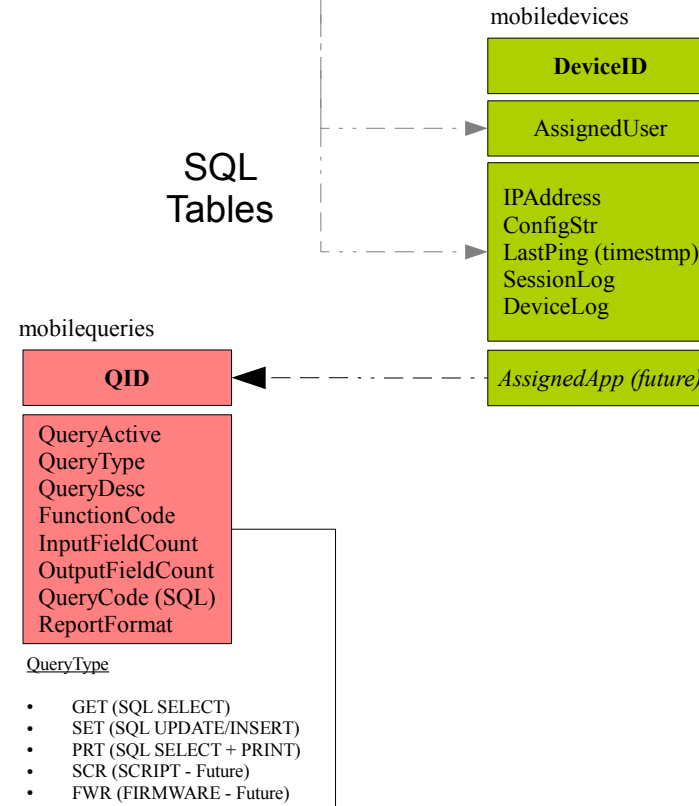
Target MRP System  
(eg b3marlborough)



Mobile Service



SQL Tables



Query Engine

Mobile Packet Parser

TCP/IP Socket 2001

Windows/UNIX Print Service

## Simple TCP Packet Message Format

Encoding ASCII. TAB-DELIMITED-DATA is ASCII chars >= 0x20 and <=0x7f (reject other chars)

Special ASCII Chars are SOH=0x01, STX=0x02, ETX=0x03, EOT=0x04, ACK=0x06, NACK=0x15

Direction	Message Details						
MOB->SVR	SOH	FnCode	STX	TAB-DELIMITED-DATA	ETX	HWRRevByte	SWRevByte EOT
Respond OK	SOH	ACK	STX	TAB-DELIMITED-DATA	ETX	EOT	
Respond ERR	SOH	NACK	STX	"ErrorCode", "Error Message"	ETX	EOT	

Note All queries require the mobile device to be assigned to a user, if not assigned then returns unassigned error  
Actual SQL Queries are stored in query table using function to lookup SQL and fields substituted real-time to form query

FnCode

GET (SELECT)

0x10	MOB->SVR	GetJob, Send Respond OK:	Field1 = JobID Field1 = JobStatusByte, Field2 = JobStatusString, Field3 = JobDescriptionString
0x11	MOB->SVR	GetAssetVC, Send: Respond OK:	Field1 = AssetID Field1 = AssetName, field2 = AssetStatus, Field3 = CurrVolume, Field4 = BatchCode, field5 = CurDip, Field6 = StatusCode
0x12	MOB->SVR	GetAssetPosn, Send: Respond OK:	Field1 = AssetID Field1 = AssetName, field2 = AssetStatus, Field3 = Posn, Field4 = Locn
0x13	MOB->SVR	GetAssetSCADA Respond OK	Field1 = AssetID Field1 = CurTemp, field2 = CurSetpoint, Field3 = Valve Status, field4 = agi Status
0x14	MOB->SVR	GetMovement, Send: Respond OK	Field1 = JobId, Field2 = AssetID Field1 = MovementID, field2 = AssetName, Field3 = AssetStatus, Field4 = EstiDip, Field5 = iDip, Field6 = EstfDip, Field7 = fDip, Field8 = fVol, Field9 = Num Additions, field10 = PadlockCode
0x15	MOB->SVR	GetAdditions, Send: Respond OK	Field1 = MovementID Field1 = SMID1Desc, Field2 = SMID2Desc, Field3 = SMID3Desc, Field4 = SMID4Desc, field5 = SMID5Description
0x16	MOB->SVR	GetSMID, Send: Respond OK	Field1 = SMID (from product stock label barcode ID scan) Field1 = ProductName, Field2 = AddQty (Incl UOM), Field3 = Addn Status

SET (EXECUTE)

0x81	MOB->SVR	SetJobInProgress, Send:	Field1 = JobID, Field2 = WiFi Password
0x82	MOB->SVR	SetJobComplete, Send:	Field1 = JobID, Field2 = WiFi Password
0x92	MOB->SVR	UnquarantineAsset, Send:	Field1 = AssetID, Field2 = WiFi Password
0x93	MOB->SVR	SetAssetLocn, Send:	Field1 = AssetID, Field2 = NewLocn, Field3 = NewPosn, Field4 = WiFi Password
0x94	MOB->SVR	SetAssetCleaned, Send:	Field1 = AssetID, Field2 = WiFi Password
0x95	MOB->SVR	SetAssetQuarantine, Send:	Field1 = AssetID, Field2 = WiFi Password
0xA0	MOB->SVR	SetInitialDip, Send:	Field1 = MovementID, Field2 = iDip, Field3 = WiFi Password
0xA1	MOB->SVR	SetFinalDip, Send:	Field1 = MovementID, Field2 = iDip, Field3 = WiFi Password
0xB0	MOB->SVR	SetAdditionComplete, Send:	Field1 = SMIID, Field2 = WiFi Password

All SET functions respond with generic OK (ACK) no fields set, wifi password is used as auth/confirmation

PRT (Print Report/label)

0xe0	MOB->SVR	Print Asset label, Send:	Field1 = AssetID, Field2 = PRINTERNAME (Optional)
0xe1	MOB->SVR	Print Tank VC Label, Send:	Field1 = AssetID, Field2 = PRINTERNAME (Optional)

System is stateless, ie each packet should be a self contained message not relying on information from previous packet.

