The new EPICS SNMP support (new devSnmp)

1. **What is new?**
    a. The new devSnmp is EPICS OSI based. So it could work on any platform which is supported by both EPICS and net-snmp;
    b. The new devSnmp supports multi versions of SNMP protocol. There are many versions of SNMP protocol, but only three versions are widely used: V1, V2c and V3. Net-snmp supports these three versions only. The new devSnmp supports V1 and V2c. The support of V3 is under consideration due to security reason.
    c. The new devSnmp is a multi-thread application. It uses thread-safe net-snmp APIs. Every SNMP agent is handled by a unique thread. So the delay led by dead/non-exist agent won't be propagated to other thread.
    d. To save system resource and network traffic, the new devSnmp opens only one session per SNMP agent. And it combines multi variables into one query. For example, if user wants to query 20 SNMP agents, 30 variables each agent, user will have 600 EPICS records. The old devSnmp creates 600 sessions and sends 600 queries. Especially if the agent name is not IP address, there will be another 600 DNS transactions. This will put significant load on network and agents. The new devSnmp creates only 20 sessions, and send only 20 queries. Each query includes 30 variables. If agent name is not IP address, there will be only 20 DNS transaction during initialization. No DNS request is needed during operation.
    e. Records are processed asynchronously. When a record is processed, it sends a request to the thread which handles the SNMP agent for it. The thread combines the request with all other requests to the same agent together and sends only one query message. Once the thread gets the response or timeout, it callbacks the record process. So the query is on-demand. The old devSnmp processes records synchronously. A separate task keeps polling SNMP agents with 20ms interval between each query. This wastes the bandwidth and can only send 50 queries per second. So it is not scalable.
    f. Bugs are fixed such as alarm process, response/parsing exception handling, data conversion …
    g. The make migration easier, the new devSnmp can take the EPICS db file and st.cmd file for old devSnmp without any modification.

2. How to use it?

a. The new devSnmp release comes with two apps: *devSnmpApp* and *snmpApp*.
   i. *devSnmpApp* is meant to be a shared module which generates a shared library *libdevSnmp.a* and *libdevSnmp.so*, user can make his/her own IOC application which links *libdevSnmp*.
   ii. *snmpApp* is an IOC application. It generates an executable file *snmp*. Under *iocBoot/iocsnmp*, the *st.cmd* is using *snmp* to start an IOC.

b. *iocBoot/iocsnmp/st.cmd* is an example for your own *st.cmd*. It follows the regular procedure to start a soft IOC. You might notice there are four special function calls:
   i. *SNMP_DRV_DEBUG(int debuglevel)* sets the driver debug level, parameter 0 will disable the driver debug info. If you don't call it, the default debug level is 0;
   ii. *SNMP_DEV_DEBUG(int debuglevel)* sets the device support debug level, parameter 0 will disable the device support debug info. If you don't call it, the default debug level is 0;
   iii. *snmpMaxVarPerMsg(int N)* specifies up to *N* variables could be combined into one query message. Each query is a UDP message. Different SNMP agent might have different size limit of the query UDP message. *N* is shared by all the SNMP agents who will be queried by the soft IOC. So *N* must be the minimum number of all SNMP agents. If you don't call this function, the default is 30.
   iv. *epicsSnmpInit()* is a legacy call from the old devSnmp. In the new devSnmp, it does nothing. We still have it here just to make the *st.cmd* backwards compatible.

c. *snmpApp/Db/snmpDemo.db* is a demo EPICS database.
   i. The new devSnmp support ai, longin, stringin and waveform record.
   ii. The DTYP could be "Snmp", "SnmpV1" and "SnmpV2c". "Snmp" is equivalent to "SnmpV2c".
   iii. The input field should be "*@host community objectId*". The db for the old devSnmp might have input like "*@host community objected mask cnt*". The new devSnmp accepts it but just ignore *mask* and *cnt*.