

SCAN, GPnP, GNS und Co. Erfahrungen mit den neuesten RAC Features

Dr. Günter Unbescheid
Database Consult GmbH
Jachenau

Schlüsselworte

Real Application Cluster, SCAN Listener, GPnP, GNS, RAC

Einleitung

Das aktuelle Release 11gR2 wartet mit einer Reihe von Neuerungen im Bereich von Real Application Clustern im Allgemeinen und Oracle Clusterware im Besonderen auf. "Single Client Access Name", "Grid Plug and Play" und "Grid Naming Service" sind einige der Schlagworte in diesem Zusammenhang. Die neuen Techniken sollen die Erweiterung von Cluster-Umgebungen durch neue Server erleichtern, indem virtuelle IP-Adressen und ein zentraler Access Name automatisiert auf dergleichen Änderungen reagieren können. Die genannten Features haben einen Umbau und eine Erweiterung der Prozessarchitektur von Oracle Clusterware nach sich gezogen. Den Vereinfachungen im laufenden Betrieb steht damit eine zunächst komplexere Grundkonfiguration gegenüber, weil DNS und DHCP-Dienste für GNS zwingend notwendig sind.

Der Vortrag stellt die unterschiedlichen Konfigurationsmöglichkeiten der genannten Features vor und geht auf die ersten praktischen Erfahrungen ein. Die grundlegende Frage hierbei ist: Wann lohnt sich der Einsatz von GPnP und wann ist es empfehlenswert eine "manuelle" Konfiguration - ohne GNS - zu implementieren.

Der Rückblick in die 10g-Welt

Stand der Technik in diesem Beitrag ist 11g Release 2 Patch Set 2 (11.2.0.3), das zum Zeitpunkt der Erstellung dieses Dokuments für Linux x86 und Linux x86-64 erschienen ist. Dieser Patch Set wird – wie neuerdings üblich – als Komplettpaket (*full installation*) über MOS verteilt und idealerweise in einem neuen, separaten Home-Verzeichnis installiert (*out-of-place upgrade*).

Im RAC-Umfeld nutzen bekanntlich mehrere Oracle-Instanzen eine Datenbank. Die entsprechenden Datenbank-Dateien müssen *shared* im Zugriff sein. Die Instanzen werden über ein separates *private interconnect* mit eigenen NICs (*private network*) synchronisiert. Der applikationsseitige Zugriff auf die Datenbanken erfolgt hingegen über eingetrenntes *public network* mit eigenen NICs und entsprechenden IP-Adressen. Alle Komponenten sollten redundant ausgelegt werden. Kommen NAS-Storagekomponenten zum Einsatz, so ist hierfür ein weiteres, getrenntes Netzwerk dringend zu empfehlen.

Netzwerktechnisch besteht jedes Datenbank-Cluster – ob 10g oder 11g – zunächst aus den folgenden Komponenten:

- IP-Adressen für das öffentliche Netzwerk, die für jeden Knoten statisch vergeben werden.
- Für jeden Knoten existieren ebenso – im gleichen Subnetz wie die statischen – seit der Version 10g virtuelle IP-Adressen. Diese Adressen werden im Fehlerfall einem „überlebenden“ Knoten zugeordnet. Der Client muss auf diese Weise beim Verbindungsaufbau nicht auf den – stark

verzögerten – TCP-Timeout warten und kann über die virtuelle Adresse den Listener zügig erreichen. Die virtuellen IP-Adressen sind – über eine entsprechende DNS-Konfiguration – VIP-Namen zugeordnet, beispielsweise *rac01-vip* für den Knoten *rac01*.

- IP-Adressen für das *private interconnect*, die ebenso pro Knoten zu vergeben sind.

Nachfolgend ist Beispiel einer Listener.ora Datei im Stile von 10g unter Verwendung der virtuellen IP-Adressen abgebildet. Der Parameter IP=FIRST sorgt dafür, dass der Listener den *listening endpoint* nur auf der IP-Adresse aufbaut, die dem unter HOST angegebenen Hostnamen entspricht. Die – hier auskommentierte – *boot* IP-Adresse kann für administrative Zwecke genutzt werden.

```
LISTENER_rac01 =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1))
      (ADDRESS = (PROTOCOL = TCP) (HOST = rac01-vip) (PORT = 1521) (IP = FIRST))
#      (ADDRESS = (PROTOCOL = TCP) (HOST = rac01) (PORT = 1521) (IP = FIRST))
    )
  )
```

Die init.ora Parameter *local_listener* und *remote_listener* müssen darüber hinaus mit Hilfe von NET-Aliasnamen so konfiguriert werden, dass die Datenbank-Services auf den Listener-Prozessen sämtlicher RAC-Knoten registriert werden können

Der Client sollte darauf aufbauend die den virtuellen Adressen zugeordneten Namen für den Verbindungsaufbau nutzen. Das folgende Beispiel zeigt einen Ausschnitt aus der Datei *tnsnames.ora*:

```
test =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = rac01-vip) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = rac02-vip) (PORT = 1521))
    (LOAD_BALANCE = yes)
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = test.DBC.de)
    )
  )
```

Bei Erweiterungen des Clusters durch zusätzliche Knoten sind aus Sicht der Netzwerkverwaltung folgende Arbeiten für jeden Server auszuführen:

- Vergabe einer statischen, öffentlichen IP-Adresse für den Knoten
- Vergabe einer statischen, öffentlichen, virtuellen IP-Adresse für den Knoten
- Vergabe einer statischen, privaten IP-Adresse für den Knoten

Diese IP-Adressen sind entsprechend im DNS zu konfigurieren.

Die Erweiterungen unter 11gR2 im Überblick

RAC 11gR2 hat sich im Kontext dieses Beitrages folgende Verbesserungen und Erleichterungen (*ease of use*) auf die Fahnen geschrieben:

- SCAN (*single client access name*) bietet eine eindeutige Adressierungsmethode für jedes Cluster, unabhängig von der Anzahl der bestehenden, neu hinzugefügten oder gelöschten Knoten.

- „Grid Plug and Play“ (GPnP) besteht im Wesentlichen aus dem Grid Naming Service (GNS), der für die dynamische Vergabe virtueller IP-Adressen und die Zuordnung der entsprechenden VIP-Namen zuständig ist.

Ziel dieser Erweiterungen ist es, die Knotenkonfiguration im RAC zu vereinfachen. Die oben genannten Features lassen sich im Detail wie folgt beschreiben

SCAN

SCAN ist ein virtueller Name, der auf einem oder – laut Empfehlung – bis zu 3 virtuellen IP-Adressen des *public* Network basiert – unabhängig davon welche Knoten und Services in dem betreffenden Cluster laufen. Über diesen Namen kann der Client mit Hilfe der zugeordneten virtuelle Adressen mit einem der verfügbaren SCAN-Listener des Clusters verbunden werden. Bei diesen SCAN-Listnern sind sämtliche Datenbank-Services des Clusters registriert. Diese Listener sorgen auch für die Weiterleitung der Client-Requests an einen herkömmlichen Listener (*load balancing*), der wie gewohnt die weitere Verbindung aufbaut. Beim Ausfall eines Knotens werden dort laufende SCAN-Listener entsprechend migriert.

SCAN-Namen können automatisiert über GNS in Verbindung mit DHCP oder manuell über DNS verwaltet und aufgelöst werden. Aus Gründen der Hochverfügbarkeit sollten jedem Scan-Namen mindestens drei IP-Adressen zugeordnet werden (*round robin access*), die alle auf dem gleichen Netzwerk wie die öffentlichen Adressen liegen müssen.

Für jede SCAN-IP-Adresse werden im Rahmen der Installation von Grid Infrastructure jeweils folgende Ressourcen angelegt: Ein individueller virtueller Name und ein Scan-Listener, der über diesen Namen erreicht wird. Dieses Konglomerat wird – wie erwähnt – über die im vorhinein konfigurierte IP-Adresse erreicht. Nach der erfolgreichen Installation können die genannten Ressourcen wie folgt kontrolliert werden:

```
[oracle@rac01] srvctl config scan_listener
SCAN Listener LISTENER_SCAN1 exists. Port: TCP:1521
SCAN Listener LISTENER_SCAN2 exists. Port: TCP:1521
SCAN Listener LISTENER_SCAN3 exists. Port: TCP:1521
[oracle@rac01] srvctl config scan
SCAN name: test-scan, Network: 1/133.22.67.0/255.255.255.0/
SCAN VIP name: scan1, IP: /sales1-scan.example.com/133.22.67.192
SCAN VIP name: scan2, IP: /sales1-scan.example.com/133.22.67.193
SCAN VIP name: scan3, IP: /sales1-scan.example.com/133.22.67.194
```

Wenn in einem Cluster mit zwei Knoten – wie oben empfohlen – drei SCAN-IP-Adressen vorgesehen werden, laufen durchaus zwei SCAN-Listener auf einem Knoten.

Da für SCANS wie erwähnt bis zu 3 IP-Adressen verwendet werden sollten, ist die Konfiguration im Rahmen von DNS notwendig, Einträge in die hosts-Datei dagegen nur möglich, wenn mit einer IP-Adresse gearbeitet wird.

Standardmäßig wird der SCAN-Name aus dem Namen des Clusters und der Domäne wie folgt zusammengesetzt: <clustername>-scan.domain, also beispielsweise mycluster.scan.DBC.DE.

Die jeweilige Datenbank registriert sich über den Parameter *remote_listener* bei den SCAN-Listnern. Der Parameter *local_listener* hingegen weist, wie auch unter der Version 10g auf den Node-spezifischen lokalen Listener-Prozess.

Der Client kann dann den NET-Aliasnamen entsprechend einstellen:

```

test =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = test-scan.DBC.DE) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = test)
    )
  )
)

```

Auch Verbindungen über Easy Connect sind wie gewohnt möglich, sofern die Datei sqlnet.ora entsprechend eingestellt wurde:

```
sqlplus system/manager@test-scan:1521/testservice
```

Die Nutzung von SCAN-Listnern ist empfohlen jedoch kein Muss.

GPnP und GNS

Auch GNS arbeitet im Kontext von Grid Infrastructure. GNS sorgt für die dynamische Bereitstellung von virtuellen IP-Adressen des public network, die für SCAN-Listener und für die Cluster-Knoten benötigt werden und die Zuordnung der entsprechenden Namen per multicast Domain Name Service (mDNS). Dem entsprechend ist GNS auf Dynamic Host Configuration Protocol (DHCP) angewiesen. GNS selbst wird über eine eigene, virtuelle IP-Adresse erreicht, die statisch im DNS konfiguriert wird, und delegiert die Adressvergaben über eine DNS-Subdomain, die vor der Installation entsprechend eingerichtet werden muss.

Auf diese Weise wird die namentliche Zuordnung von virtuellen IP-Adressen vom DNS in die Clusterware verlagert, die aus technischer Sicht einen eigenen DNS-Service zur Verfügung stellt (*subdomain delegation*). Bei der Installation sind in diesem Konzept nur die *boot* IP-Adressen für das öffentliche und das private Netz vorzugeben, den Rest erledigt GPnP. Dadurch soll im späteren Betrieb das Hinzufügen und Abändern von Clusterknoten erleichtert werden. Allerdings werden mit GPnP den DBAs Aufgaben zugesprochen, die in herkömmlichen Umgebungen außerhalb ihres Verantwortungsbereiches liegen.

Zur Vorbereitung der Installation ist im DNS eine entsprechende Subdomain – beispielsweise *grid.DBC.DE* – vorzubereiten, die entsprechende „Einstiegs“-IP-Adresse festzulegen sowie im DHCP für diese Domäne ein Bereich von möglichen IP-Adressen zu reservieren.

Konfigurationshinweise

Auf den Vortragsfolien dieses Beitrags werden die Details der Konfiguration für SCAN und GNS ausführlich dargestellt. Hierbei werden folgende Beispiele ausgearbeitet: SCAN-Konfiguration ohne GNS sowie GNS und SCAN im Verbund.

Zusammenfassung

Die dargestellten Features bieten zweifelsohne interessante Möglichkeiten dynamischer Knotenerweiterungen im RAC. Vor allem im Zusammenspiel mit GNS werden Datenbank-Verantwortliche jedoch mit umfangreichen Konfigurationsaufgaben betraut, die in vielen Fällen das gewohnte Aufgabenspektrum deutlich erweitern dürften.

Hinzu kommt, dass dynamische Ressourcenanpassungen auch auf anderen Ebenen möglich sind, beispielsweise durch das Lastbedingte Starten von DB-Services auf den Instanzen des RAC oder das Starten neuer Instanzen auf bereits vorbereiteten RAC-Servern.

Kontaktadresse:

Dr. Günter Unbescheid

Database Consult GmbH

Laich 9 1/10

D-83676 Jachenau

Telefon: +49 (0) 8043 1010

Fax: +49 (0) 8043 1011

E-Mail g.unbescheid@database-consult.de

Internet: www.database-consult.de