

HOW TO DEAL WITH AN INHERITED SQL SERVER

INTRODUCTION

Whether you're a newly minted DBA or a highly experienced one, the first week at a new job can stretch your patience to breaking as you try to get answers to a wide range of questions about how things work and why things are done (or not done) a certain way. It is perhaps the most dangerous week you'll have as a DBA, because you don't know exactly what you don't know. This document is meant to be a roadmap for that first week to help you make sure you're asking the right questions and prioritizing your efforts in a way to minimize risk. It's also to help you meant to help you set expectations for yourself and those you work with.

TASK #1 TAKE NOTES, TRIAGE, AND DO NO HARM

You're going to have a lot of questions in the first week and the weeks that follow. Write them down and to log the answers as you get them. You're also going to see things that look wrong—things that make you itch to fix them. Write those down too. This list will help you brief your employer on what you're doing and what you've learned. It's also going to help you prioritize how to spend your time during the next few weeks. In addition, these notes can serve as the basis for documentation for the next DBA.

Don't rush to "fix" anything except backups and problems that are preventing people from working. There are two reasons for this approach:

The first is you'll want to do a thorough initial survey in order to prioritize effectively. If you jump right into fixing the first problem you identify, it may delay finding and fixing other more critical problems.

The second reason is things may not be as "broken" as you think. For example, I've worked in environments that deliberately did not backup the DEV and QA environments at all, and I've seen servers so heavily loaded, executing a log backup caused wait times to increase. Remember to be practical. They've gotten this far with things just as they are; they can stand to wait a bit longer.

TASK #2 SETUP YOUR WORKSTATION

Back in the dark ages, data was kept on punch cards and magnetic tape files. The physical data and logical data. The week starts with getting access to a machine and to the domain. Typically, you'll be given a freshly-imaged laptop. It's going to take a few hours to set it up the way you want, and you can work on this task throughout the week – but it should be done by the end of the week. Getting this task done can be even harder if you don't have local admin access. Have your list ready before Day 1 so you only have to make one request to load the stuff you need.

- **Encryption Software:** This should be loaded for you as a pre-boot driver. If not, see if the hard drive has an encryption option; see if Bit Locker can be enabled, or install TrueCrypt. You do not want to be the employee who loses a laptop that is not encrypted!
- **Tag it:** Tape a business card to your laptop or write your phone number on masking tape. If you do misplace it, you want an easy way for people to contact you.
- **Email and Office:** This also should be loaded for you, and it's a must-have. It may also take some time for your email account to be provisioned and for you to receive a license key. At the very least, you need Outlook, Word, and Excel (or their equivalents). Visio is always nice to have. You may need InfoPath and Project (or the Project viewer).
- **SQL Server:** At a minimum, you need Management Studio, but my preference is to load two complete instances of the Developer Edition of the most recent version of SQL Server. Set them up in mixed mode with a strong SA password and apply the latest service pack. This gives you a practice environment you completely control.
- **VPN Client:** You need the client and the configuration for it (often pre-loaded). Determine if VPN access requires two factor authentication (RSA token or similar), then Test the VPN from a remote network as soon as you can—don't wait until you need it to find out you don't have access!
- **Other SQL Tools:** Find out what commercial tools your employer uses for managing and monitoring the SQL environment—then find the install package and license key for each. Some of these require connecting to a repository you'll have to identify.

- **Task/Project Management/Time Clock/Change Management:** You need the application or URL to see what you should be working on, project plans, and where/how to enter your time.
- **Source Control:** You need both the client and the location of the repository.
- **Miscellaneous Utilities:** I recommend, at a minimum, Notepad++ with an XML plugin, Remote Desktop Connection Manager, 7Zip, and ConEmu.
- **Free SQL Tools:** Most vendors that build SQL Server tools have one or more free tools—a list of my favorite free tools from Idera is provided at the end of this document.
- **Anti-Virus and Fully Patched:** Obvious and usually done for you, but critical so double-check.

TASK #3 GETTING ACCESS

This seems like it should be easy: receive domain credentials and attach to the DBA group. In practice, however, there may be any number of resources to which you won't initially have the access you need.

- **SQL:** The obvious one — you **MUST** be able to connect to your servers as a member of Sysadmin.
- **Firewall:** In many environments, your IP address will need to be added to a firewall policy.
- **SharePoint:** There is usually an IT and/or DBA site which requires the site admin to grant you access.
- **Admin Login:** Best practice is to have one login for your machine and another for connecting to servers.
- **VPN:** Often requires you to be added to another group and sometimes a two-factor token or grid card
- **Terminal/Jump Box:** Some environments won't allow you to connect directly to production servers; you have to remote desktop to an intermediate workstation.
- **Server Access:** This is nice to have, but often not granted to DBAs.
- **Source Control:** You may need to retrieve SSIS packages—maybe even DDL updates from source control.
- **Other Monitoring/Management Tools:** Depending on the organization, you may get read-only access to tools to monitor or manage your SAN, servers, virtual machines, and third-party backup solutions.

You may also need to be added to various distribution lists. It's hard to know what those are until hear about an email you didn't get during a meeting!

TASK #4 FIND THE DOCUMENTATION

When we were first creating the SQL language, we added simple aggregate functions as far back as the I know saying “documentation” might generate a cynical grin as you read this, but you don’t want to fall into the trap of not looking. The best place to start is the DBA SharePoint site or a DBA network share. Your goal is to have a couple of places to look for answers before you start emailing questions. Ideally, you want to find documents to cover all of these categories:

- **Change Management**
- **Disaster Recovery**
- **Server Build/Provisioning**
- **Calendar of Planned Maintenance Windows**
- **Service Level Agreements**
- **Database Maintenance**
- **Data Dictionaries**
- **Server and Network Diagrams Related to SQL**
- **Security Policies**
- **Backup Policy**
- **Ticket/Escalation Process**
- **Patching Process**

Keep track of each one you find, and any questions you have about each document. You should also make a note of any document you can’t find—this is work that needs to get done at some point over the next few weeks.

TASK #5 IDENTIFY YOUR INSTANCES

Now you’re getting to the good stuff! It’s vital that you understand what servers you manage and to what extent you manage them. The best case is a Central Management Server (CMS) already set up— you just have to figure out where it is. Next best is a list of registered servers on a terminal or ask another DBA to export their list and send it to you. If all else fails, you see how many servers Management Studio can identify and start from there.

It’s easy to get tangled up here if you don’t get good guidance. There may be servers on multiple domains. There may be servers that you don’t administer, or only administer lightly. There may be instances that don’t broadcast their availability. There may also be multiple instances and multiple versions per server. Nothing should scare you more than the idea of servers you do not know about. This is one more reason to delay all but the most critical work until you’re absolutely certain you have a full view of the environment.

Once you have what you think is the complete list, you need to do two things: get someone to confirm the list and then tag each server as DEV or QA or PROD (and ideally your CMS should match that). A deeper inventory of versions and other data can come in later weeks, but you must have this figured out by the end of the first week.

I like to take time to set up Remote Desktop Connection Manager at the same time. It’s not often I need to remote connect to the server, but when I do, I like to see the servers organized the same way as in Management Studio.

TASK #6 CHECK THE BACKUPS AND BACKUP STRATEGY

There is nothing more important than a backup of your data and nothing more important to verify in your first week—after that you own it! I can't emphasize enough your task is to validate what your employer thinks is happening is really happening. Plan to only make critical changes to backup plans in the first week, and I define critical as "database/s management wants backed up is not currently backed up".

The first question is to determine how the backups are executed. Most of the time it will be a SQL Agent job with a reasonably obvious name and that's good; it's something you can see and manage easily. If you don't see backup jobs then you have to dig deeper; it could be a Windows Task, it could be a homegrown job system running on another server, it could be a third-party backup that uses an agent, or it could be there is no scheduled backup. Usually, this will be the same method across all instances—but check to be certain.

Next, we want to review the recovery models. There is no telling what pattern you'll find—it could be full everywhere, full only in production, full only on certain servers, etc. It could be more fine-grained where time has been taken to determine simple recovery is good enough on a case-by-case basis. Even if there is supposed to be a pattern, it may not have been enforced. You may see recovery models you don't agree with—note them for future discussion and move on. The most important thing to understand is whether you need log backups and, if so, of what databases, so you can verify they are occurring. They have to happen so you have a solid recovery strategy, and they have to happen so you don't run out of disk space.

Next, you need to make sure jobs are being executed successfully and include all the databases. This means checking the job history – you should see a few days or more of successful runs. If you don't see history, you'll have to research more—find out if it's because the history is getting removed quickly, or if the job isn't running—this could be a disabled job, a disabled schedule, or even a problem with SQL Agent. Open each job (and the maintenance plan, if it uses one) to see exactly what it is doing. Just because a job is titled "Backup All Databases" doesn't mean it has not been modified to exclude one or more databases.

NOTE: It is entirely possible some databases or even whole instances are considered of such low value they don't get backed up. While this does not feel right to us as DBAs, if you do run into this, write it down and come back to it later, but don't get caught up in arguing for a change—there will be plenty of time to address this later.

Finally, you find out how many days of backups you are supposed to be keeping locally and who is responsible for getting those backups onto tape (usually you'll backup to disk and then disk to tape, but it could be direct-to-tape). This is also the time to find out who to call if you need a tape restored, and how long it takes to get an offsite tape. Don't wait until you need to restore to find out.

TASK #7 CHECK ON FREE SPACE AND FILE GROWTH

If there is one thing that can jump up and bite you unexpectedly, it's running out of space. This might be running out of space in a database file, a log file isn't set to auto-grow, or the drive itself runs out of space. This is another instance where you want to see if there is a particular reason for the current configuration before you start implementing changes. It's definitely worth taking the time to determine if there is an existing system in place already to monitor free space —check with the server team too!

TASK #8 LEARN THE CHANGE MANAGEMENT PROCESS & MAINTENANCE WINDOWS

No two companies handle change management the same way. Some are very informal: a verbal request or an email. In others, all changes go through a review and approval process and have to include a plan for the change as well as a plan to roll back the change if needed. There are often changes that can be made outside of change management based on a defined list or the DBA's appraisal of the risk.

As the new DBA you want to be able to answer the question “how do I know it's ok to apply this change?” Is it based on process, a sign-off from your manager or a project manager, or just your judgment? What should a change request look like? Should it be a complete script that runs or fails, or is it more like “set up a new database for client X?”

Directly related to change management, is understanding when changes can be made. It might be as restrictive as one Sunday morning per month, or as liberal as allowing changes to be made as they come in—though usually it's somewhere in between. There is often a separate time devoted to maintenance when DBA-type tasks can be completed—things like index and statistics rebuilds, but also work that requires the database or instance to be offline.

Don't be surprised if you hear something other than a perfectly clear answer. That's fine; plan to confirm all your actions with your team or your manager during the first few weeks—until you've absorbed the intricacies of what's acceptable.

TASK #9 CHECK ON SERVICE PACKS

This is an easy one. You want to run a quick inventory per instance to get the version and service pack level. You will probably find some are not on the latest service pack. This may be because they were missed, or maybe your employer has a reason for not applying it—find out which before taking action.

Don't expect to fix this in the first week—or even in the first month. Most companies want to run the service pack in DEV and QA before applying to production. Applying a service pack seems easy, but it requires a lot of preparation outside of the DBA team and it involves risk. You want to be settled in before you start this so you are familiar with the environment and your colleagues know and have confidence in you. It may also not be the most critical thing you need to fix. So why do I recommend you check on something that you shouldn't expect to fix right away? Two reasons. First, not being up-to-date on patches is a security and compliance risk, and second, it's a really good indicator of the maturity of the database environment. It's great to find everything is up-to-date and, while this doesn't mean everything actually is, it is a good indicator. Conversely, if you find that the service packs have not been applied, then you should think more about other tasks that may not be getting done.

BONUS! FREE TOOLS FROM IDERA

The following is a list of recommended free tools available from IDERA — many of these made it onto my “must-have” list of SQL Server management tools:

SQL Check Real-time performance monitoring for SQL Servers

SQL Integrity Check Identify SQL Server corruption

SQL Fragmentation Analyzer Detect fragmentation hotspots across SQL Servers

SQL Permissions Extractor Extract and migrate permissions from objects in SQL

SQL Job Manager View & manage SQL Server jobs across multiple servers

SUMMARY

Whether you're a newly minted DBA
The tasks outlined above are based on
my own experiences during my first
weeks as a DBA new to an organization
and form the basis for my own
“best practices” during my first weeks –
follow these guidelines and you'll
be off to a great start. They may well
help you identify follow up tasks that
will take weeks to complete.
Remember to prioritize your efforts
so that you do the things that matter
most first.

ABOUT THE AUTHOR

Mr. Andy Warren is a SQL Server consultant based in Orlando. Andy was a founding member of SQLServerCentral.com, President of the Orlando SQL Server Users Group, and one of the founders of SQLSaturday. He has been an MVP since 2009 and has written and presented extensively on topics relevant to SQL Server professionals.

Andy blogs at <http://sqlandy.com>, is on LinkedIn at <http://www.linkedin.com/in/sqlandy>, and is occasionally on Twitter as @SQLAndy.

IDERA understands that IT doesn't run on the network – it runs on the data and databases that power your business. That's why we design our products with the database as the nucleus of your IT universe.

Our database lifecycle management solutions allow database and IT professionals to design, monitor and manage data systems with complete confidence, whether in the cloud or on-premises.

We offer a diverse portfolio of free tools and educational resources to help you do more with less while giving you the knowledge to deliver even more than you did yesterday.

Whatever your need, IDERA has a solution.

I D E R A

SQL INVENTORY MANAGER

DISCOVER, TRACK AND MANAGE YOUR SQL SERVER INVENTORY

- View your SQL Server inventory - know what you have where & who owns it
- Auto-discover any new servers installed, to better manage server sprawl
- Get alerts when a server goes down, or space is running low
- Set tags to better organize servers and databases across the enterprise
- Quickly deploy and access from anywhere via web-based and agentless UI

Start for FREE

The screenshot displays the SQL Inventory Manager dashboard. At the top, it shows the user 'dev/administrator' and navigation links for 'LOG OUT' and 'HELP'. The dashboard is divided into several sections:

- Health Check Recommendations:** A list of 29 recommendations with severity levels. Key items include:
 - LEVEL 3: 2 instance(s) are down.
 - LEVEL 3: 24 databases do not have a backup.
 - LEVEL 2: Instance AUT-2005-001\SQL2005 configured to allow xp_cmdshell use.
 - LEVEL 1: 2 instances not set to optimize ad hoc workloads.
- TOP DATABASES BY SIZE (MB):** A bar chart showing 'tempdb (D)' as the largest database at 610.1 MB.
- TOP DATABASES BY ACTIVITY (TRANSACTIONS/SEC):** A bar chart showing 'tempdb (A)' and 'tempdb (D)' as the most active databases.
- INSTANCES (6 instances | 54 databases | 939.4 CB total size):** A table listing instances with their status, response time, version, and size of databases.

SQL Server	Status	Response Time	Version	# of DBs	Size of DBs (MB)
DEVPERFOT\SQL2005	Down	-1 ms	SQL Server 2005 SP4 (9.0.5000.0)	26	961391.2
DEVPERFOT\SQL2008R2	Down	-1 ms	SQL Server 2008 R2 RTM (10.50.1600.1)	0	0.0
AUT-2005-001\SQL2005	Up	1 ms	SQL Server 2005 RTM (9.0.1399.6)	4	24.1

On the right side, a 'MY ENVIRONMENT' summary shows:

- Instances: 6
- Servers: 4
- Databases: 54
- Data Size: 229.5 GB
- Log Size: 710 GB
- Disabled Instances: 0
- Available Licenses: Unlimited

Below this, 'TAGS' are listed, including 'Production', 'Disaster Recovery', 'Development', 'Business Critical', '24x7', and 'Non-Critical'.