

# Table of Contents

<b>Preface</b>	<b>1</b>
<b>Chapter 1: Understanding Replication Concepts</b>	<b>7</b>
<b>The CAP theory and physical limitations</b>	<b>7</b>
Understanding the CAP theory	8
Why the speed of light matters	9
Long distance transmission	10
Why latency matters	10
<b>Different types of replication</b>	<b>10</b>
Synchronous versus asynchronous replication	10
Understanding replication and data loss	12
Considering the performance issues	12
Single-master versus multi-master replication	13
Logical versus physical replication	14
When to use physical replication	15
When to use logical replication	15
<b>Using sharding and data distribution</b>	<b>16</b>
Understanding the purpose of sharding	16
An example of designing a sharded system	16
An example of querying different fields	17
Pros and cons of sharding	19
Choosing between sharding and redundancy	20
Increasing and decreasing the size of a cluster	20
Combining sharding and replication	22
Various sharding solutions	24
PostgreSQL-based sharding	24
External frameworks/middleware	24
<b>Summary</b>	<b>25</b>

<b>Chapter 2: Understanding the PostgreSQL Transaction Log</b>	<b>27</b>
<b>How PostgreSQL writes data</b>	<b>27</b>
The PostgreSQL disk layout	28
Looking into the data directory	28
PG_VERSION – PostgreSQL version number	29
base – the actual data directory	29
global – the global data	31
pg_clog – the commit log	31
pg_hba.conf – host-based network configuration	31
pg_ident.conf – ident authentication	32
pg_multixact – multi-transaction status data	32
pg_notify – LISTEN/NOTIFY data	32
pg_serial – information about committed serializable transactions	32
pg_snapshot – exported snapshots	32
pg_stat_tmp – temporary statistics data	32
pg_subtrans – subtransaction data	32
pg_tblspc – symbolic links to tablespaces	33
pg_twophase – information about prepared statements	33
pg_XLOG – the PostgreSQL transaction log (WAL)	33
postgresql.conf – the central PostgreSQL configuration file	34
Writing one row of data	35
A simple INSERT statement	35
Read consistency	38
The purpose of the shared buffer	39
Mixed reads and writes	40
<b>The XLOG and replication</b>	<b>41</b>
<b>Understanding consistency and data loss</b>	<b>41</b>
All the way to the disk	42
From memory to memory	43
From memory to the disk	44
One word about batteries	45
Beyond fsync()	46
PostgreSQL consistency levels	46
<b>Tuning checkpoints and the XLOG</b>	<b>48</b>
Understanding the checkpoints	48
Configuring checkpoints	48
About segments and timeouts	49
To write or not to write?	50
Tweaking WAL buffers	52
<b>The internal structure of the XLOG</b>	<b>53</b>
Understanding the XLOG records	53
Making the XLOG deterministic	53
Making the XLOG reliable	54
LSNs and shared buffer interaction	55
Debugging the XLOG and putting it all together	55
<b>Summary</b>	<b>57</b>

<b>Chapter 3: Understanding Point-In-Time-Recovery</b>	<b>59</b>
<b>Understanding the purpose of PITR</b>	<b>60</b>
Moving to the bigger picture	60
<b>Archiving the transaction log</b>	<b>62</b>
<b>Taking base backups</b>	<b>64</b>
Using pg_basebackup	65
Modifying pg_hba.conf	65
Signaling the master server	66
pg_basebackup – basic features	66
Making use of traditional methods to create base backups	69
Tablespace issues	69
Keeping an eye on network bandwidth	70
<b>Replaying the transaction log</b>	<b>70</b>
Performing a basic recovery	71
More sophisticated positioning in the XLOG	74
Cleaning up the XLOG on the way	75
Switching the XLOG files	77
<b>Summary</b>	<b>77</b>
<b>Chapter 4: Setting up Asynchronous Replication</b>	<b>79</b>
<b>Setting up streaming replication</b>	<b>79</b>
Tweaking the config files on the master	80
Handling pg_basebackup and recovery.conf	81
Making the slave readable	82
The underlying protocol	83
<b>Configuring a cascaded replication</b>	<b>84</b>
<b>Turning slaves to masters</b>	<b>86</b>
<b>Mixing streaming and file-based recovery</b>	<b>87</b>
The master configuration	87
The slave configuration	88
<b>Error scenarios</b>	<b>89</b>
Network connection between the master and slave is dead	89
Rebooting the slave	89
Rebooting the master	90
Corrupted XLOG in the archive	90
<b>Making the streaming-only replication more robust</b>	<b>90</b>
<b>Efficient cleanup and the end of recovery</b>	<b>91</b>
Gaining control over the restart points	91
Tweaking the end of your recovery	92
<b>Conflict management</b>	<b>92</b>
<b>Dealing with the timelines</b>	<b>95</b>
<b>Summary</b>	<b>96</b>

---

<b>Chapter 5: Setting up Synchronous Replication</b>	<b>97</b>
<b>Setting up synchronous replication</b>	<b>97</b>
Understanding the downside of synchronous replication	98
Understanding the application_name parameter	98
Making synchronous replication work	99
Checking replication	100
Understanding performance issues	101
Setting synchronous_commit to on	102
Setting synchronous_commit to remote_write	102
Setting synchronous_commit to off	102
Setting synchronous_commit to local	103
Changing durability settings on the fly	103
<b>Understanding practical implications and performance</b>	<b>104</b>
<b>Redundancy and stopping replication</b>	<b>106</b>
<b>Summary</b>	<b>106</b>
<b>Chapter 6: Monitoring Your Setup</b>	<b>107</b>
<b>Checking your archive</b>	<b>107</b>
Checking the archive_command	107
Monitoring the transaction log archive	108
<b>Checking pg_stat_replication</b>	<b>109</b>
Relevant fields in pg_stat_replication	109
<b>Checking for operating system processes</b>	<b>111</b>
<b>Dealing with monitoring tools</b>	<b>111</b>
Installing check_postgres	112
Deciding on a monitoring strategy	112
<b>Summary</b>	<b>113</b>
<b>Chapter 7: Understanding Linux High Availability</b>	<b>115</b>
<b>Understanding the purpose of high availability</b>	<b>115</b>
<b>Measuring availability</b>	<b>116</b>
<b>History of high-availability software</b>	<b>118</b>
OpenAIS and Corosync	119
Linux-HA (Heartbeat) and Pacemaker	119
<b>Terminology and concepts</b>	<b>120</b>
<b>High availability is all about redundancy</b>	<b>121</b>
<b>PostgreSQL and high availability</b>	<b>123</b>
High availability with quorum	124
High availability with STONITH	126
<b>Summary</b>	<b>127</b>
<b>Chapter 8: Working with pgbouncer</b>	<b>129</b>
<b>Understanding fundamental pgbouncer concepts</b>	<b>130</b>

---

<b>Installing pgbouncer</b>	<b>130</b>
<b>Configuring your first pgbouncer setup</b>	<b>131</b>
Writing a simple config file and starting pgbouncer up	131
Dispatching requests	132
More basic settings	133
Authentication	134
Connecting to pgbouncer	134
Java issues	135
Pool modes	135
Cleanup issues	136
<b>Improving performance</b>	<b>136</b>
A simple benchmark	137
<b>Maintaining pgbouncer</b>	<b>139</b>
Configuring the admin interface	139
Using the management database	140
Extracting runtime information	140
Suspending and resuming operations	142
<b>Summary</b>	<b>143</b>
<b>Chapter 9: Working with pgpool</b>	<b>145</b>
<b>Installing pgpool</b>	<b>145</b>
Installing pgpool-regclass and insert_lock	146
<b>Understanding pgpool features</b>	<b>146</b>
<b>Understanding the pgpool architecture</b>	<b>148</b>
<b>Setting up replication and load balancing</b>	<b>149</b>
Password authentication	152
Firing up pgpool and testing the setup	152
Attaching hosts	153
<b>Checking replication</b>	<b>155</b>
<b>Running pgpool with streaming replication</b>	<b>156</b>
Optimizing pgpool configuration for master/slave mode	157
<b>Dealing with failovers and high availability</b>	<b>158</b>
Using PostgreSQL streaming and Linux HA	158
pgpool mechanisms for high availability and failover	159
<b>Summary</b>	<b>160</b>
<b>Chapter 10: Configuring Slony</b>	<b>161</b>
<b>Installing Slony</b>	<b>161</b>
<b>Understanding how Slony works</b>	<b>162</b>
Dealing with logical replication	162
The slon daemon	164
<b>Replicating your first database</b>	<b>165</b>
<b>Deploying DDLs</b>	<b>170</b>

---

<b>Adding tables to replication and managing problems</b>	<b>171</b>
<b>Performing failovers</b>	<b>174</b>
Planned failovers	175
Unplanned failovers	176
<b>Summary</b>	<b>176</b>
<b>Chapter 11: Using Skytools</b>	<b>177</b>
<hr/>	
<b>Installing skytools</b>	<b>177</b>
<b>Dissecting skytools</b>	<b>178</b>
<b>Managing pgq-queues</b>	<b>178</b>
Running pgq	179
Creating queues and adding data	179
Adding consumers	181
Configuring the ticker	181
Consuming messages	184
Dropping queues	185
Using pgq for large projects	186
<b>Using londiste to replicate data</b>	<b>186</b>
Replicating our first table	187
<b>One word about walmgr</b>	<b>191</b>
<b>Summary</b>	<b>191</b>
<b>Chapter 12: Working with Postgres-XC</b>	<b>193</b>
<hr/>	
<b>Understanding the Postgres-XC architecture</b>	<b>194</b>
Data nodes	194
GTM – Global Transaction Manager	195
Coordinators	195
GTM Proxy	195
<b>Installing Postgres-XC</b>	<b>195</b>
<b>Configuring a simple cluster</b>	<b>196</b>
Creating the GTM	196
<b>Optimizing for performance</b>	<b>200</b>
Dispatching the tables	200
Optimizing the joins	201
Optimizing for warehousing	202
Creating a GTM Proxy	202
<b>Creating the tables and issuing the queries</b>	<b>203</b>
<b>Adding nodes</b>	<b>204</b>
<b>Handling failovers and dropping nodes</b>	<b>205</b>
Handling node failovers	205
Replacing the nodes	205
Running a GTM standby	207
<b>Summary</b>	<b>207</b>

---

<b>Chapter 13: Scaling with PL/Proxy</b>	<b>209</b>
<b>Understanding the basic concepts</b>	<b>209</b>
Dealing with the bigger picture	210
Partitioning the data	211
<b>Setting up PL/Proxy</b>	<b>212</b>
A basic example	213
Partitioned reads and writes	215
<b>Extending and handling clusters in a clever way</b>	<b>218</b>
Adding and moving partitions	218
Increasing the availability	220
Managing the foreign keys	220
Upgrading the PL/Proxy nodes	221
<b>Summary</b>	<b>222</b>
<b>Index</b>	<b>223</b>