



DcentraLab  
Diligence



## Audit Report

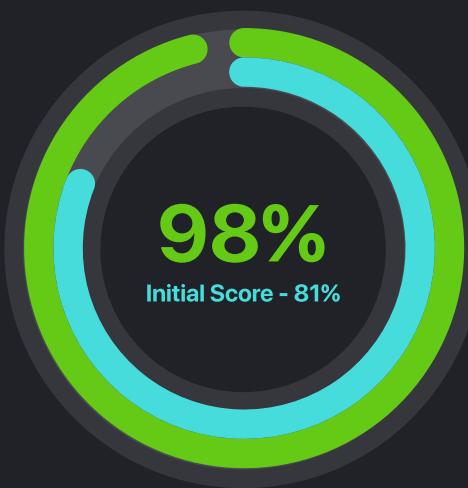
# HyperCycle - **HyperCycleFundHYPCPoolV3** **& ShareManager**

<https://www.hypercycle.ai>

# Security Audit Score

## Minimal Risk

DcentraLab Diligence team has conducted an extensive audit on HyperCycle Contracts and has found the code to be in minimal risk level given proper deployment and multi-sig permissioning.



- Minimal Risk
- Low Risk
- Medium Risk
- High Risk
- Critical Risk

Score Guidelines 

## Scope

### Audited Repository:

<https://github.com/hypercycle-development/hypercycle-contracts>

### Audited Branch:

develop

### Audited Commit Hash:

[acdc7f7aa91ba616e7c5580b62e9049b6c76dc2e](#)

### Fix Branch:

develop

### Fix Commit Hash (Iteration 1):

[9fc4f46780540d0f7997c3f2e23944e1739b7216](#)

### Audited Contracts:

- HyperCycleShareManager.sol: Audited only the differences from the previous audit report
- CrowdFundHYPCPoolV3.sol: Audited only the differences from the CrowdFundHYPCPoolV2 report

### Reviewed For Context:

- HyperCycleLicense.sol
- HYPCSwapV2.sol
- HYPC.sol
- HyperCycleShareTokensV2.sol
- ShareManagerTypes.sol
- CHYPC.sol
- CrowdFundHYPCPoolV2.sol
- HyperCycleToken.sol

### Nomenclature Of Issues:

**E** - Environmental

**A** - Contract HyperCycleShareManager.sol

**B** - Contract CrowdFundHYPCPoolV3.sol

## Contracts Architecture Overview

### HyperCycleShareManager analysis

HyperCycleShareManager is a governance contract and a management contract for the HyperCycleShareTokens contract. This contract allows the share owner to create a variety of proposals regarding its share management. It allows the shareholders to vote on those proposals on chain. The governance token is the "wealth" token which is part of the HyperCycleShareTokens contract which is a ERC-1155 implementation. The main idea of the Share Manager contract is to extend the Share Tokens contract with a governance mechanism that allows share token holders to execute management functionalities for their share by voting on proposals.

A proposal can be ended and executed by the token holders any time the consensus is reached. The share owner will be able to create votations for the following features:

- Change the hardware operator - required more than 50% consensus
- Change the hardware operator revenue - required more than 50% consensus
- Cancel share tokens - require 100% consensus
- Transfer the share ownership to a new share owner - require 100% consensus

One more important feature of Share Manager contract is the ability to migrate the Share Tokens from the Share Tokens contract to the Share Manager Contract, this will allow the Share Manager Contract to be the one holding the Share Tokens and be able to manage the Share proposals and votations. The Share owner will be able to claim the Hypc tokens based on the amount of the wealth tokens available in case the share proposal is ended and the CHyPC exists.

To migrate the Share Tokens to the Share Manager, the Share Token owner needs to call `startShareProposalMigration` function, this function will start the migration and the Share Manager will be able to finish the migration only if the ownership of the share token is changed to the Share Manager, and the Share Proposal is pending.

#### Purpose:

The CrowdFundHYPCPoolV3 contract pools HyPC tokens from multiple users to collectively create proposals and acquire c\_HyPC tokens, which can back licenses within the HyperCycle ecosystem.

#### Features:

Proposal Creation: Users create proposals with HyPC collateral, acting as interest for contributors.

Deposits and Interest: Users deposit HyPC into proposals and earn interest over terms ranging from 18 to 36 months.

## Contracts Architecture Overview

Token Swapping: Pooled HyPC is swapped for c\_HyPC tokens at various levels once proposals are funded.

Token Redemption: After the term ends, c\_HyPC tokens can be redeemed back into HyPC, allowing depositors to reclaim their investments plus interest.

Transferability: Deposits and proposals can be transferred between users, enhancing liquidity and flexibility.

### Interaction: The contract interfaces with:

CHYPC.sol: Manages c\_HyPC tokens.

HyperCycleSwapV2.sol: Handles HyPC and c\_HyPC token swaps.

### Key Components:

ContractProposal Struct: Defines proposal details, including owner, term, interest rate, and number of NFTs.

UserDeposit Struct: Defines user deposit details, including amount and associated proposal.

Modifiers and Events: Ensure valid operations and track significant actions like creation, deposits, swaps, and redemptions.

### Functions Overview:

createProposal: Validates and creates proposals, calculating required funds and interest.

createDeposit: Allows users to back proposals with HyPC deposits.

transferDeposit & transferProposal: Enable the transfer of deposits and proposals between users.

swapTokens: Swaps HyPC for c\_HyPC tokens after a proposal starts.

redeemTokens & redeemSingleToken: Redeems c\_HyPC tokens back into HyPC post-proposal completion.

## Issues Severity Reference Table

### Type

#### Discussion

The issue severity is dependent on design, centralization, and product specifications of the project.

#### Informational

This issue is not critical and does not pose an immediate threat to the functionality or security of the smart contract. It is simply an informational item that the auditors have identified and recommends addressing for best practices or to improve the overall performance of the contract.

#### Low

This issue is relatively minor and does not pose a significant risk to the functionality or security of the smart contract. While it is recommended to address these issues to ensure the highest level of quality and security, they are not likely to cause significant problems if left unaddressed.

#### Medium

This issue poses a moderate risk to the functionality or security of the smart contract. While it may not be immediately exploitable, it has the potential to cause problems in the future if left unaddressed. It is recommended to address these issues as soon as possible to prevent any potential negative impact on the contract.

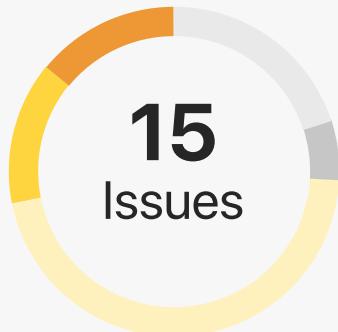
#### High

This issue poses a significant risk to the functionality or security of the smart contract. Addressing these issues as soon as possible is recommended to prevent any potential negative impact on the contract. Failure to address these issues could result in significant problems and potential loss of funds or other assets.

#### Critical

This issue poses an immediate and severe risk to the functionality or security of the smart contract. It is recommended to address these issues immediately to prevent any potential negative impact on the contract. Failure to address these issues could result in catastrophic problems and significant loss of funds or other assets.

## Findings Summary



- |   |               |   |               |
|---|---------------|---|---------------|
| ● | Discussion    | ● | Medium Risk   |
| ● | Informational | ● | High Risk     |
| ● | Low Risk      | ● | Critical Risk |

ID	Title	Severity	Status	Risk Points
B.1	Missing Validations in View Function	Low	Resolved	-1
B.2	Missing Validation on Constructor Parameter	Low	Resolved	-1
B.3	Improve Validation Logic	Informational	Resolved	
B.4	Duplicate Code	Low	Resolved	-2
B.5	Misleading Revert Message Description	Low	Resolved	-2
B.6	Missing Sanity Check for 'level' Parameter	Low	Resolved	
B.7	Duplicate Calculation	Low	Resolved	-2
B.8	Wrong 'nftsSwapped' Calculation Logic	Medium	Acknowledged	
B.9	Clarification on Token Redemption Before Proposal Completion	Discussion	Acknowledged	

## Findings Summary

ID	Title	Severity	Status	Risk Points
B.10	Incorrect 'nftsSwapped' Calculation Logic	Medium	Acknowledged	
B.11	APR computed only based on Level 10	Discussion	Resolved	
B.12	Poor Naming Convention	Low	Resolved	-2
B.13	Incorrect Token Removal in redeemSingleToken Function	High	Resolved	-15
B.14	Use of Memory Instead of Storage	High	Resolved	
B.15	Apr Calculation Only For Level 10	Discussion	Acknowledged	

## Quality Score Card

Factor	Positive Score Points range	Final Score
Presence of up to date documentation	1-5	5
Code readability	1-5	3
Operational security of the team	1-5	5
Code reliance on third-party software	1-10	10
Test Coverage	1-10	9
		<b>Total quality score: 32/35</b>

## Total Score

	Risk Score	Quality Score	Total Score
<b>Pre-Audit</b>	75 (*0.65)	32	<b>81</b>
<b>Final Report</b>	98 (*0.65)	32	<b>96</b>



## Complete Analysis

### Local Contract Findings:

Contract: HyperCicleShareManager.Sol

---

**No Issues Found**

## Complete Analysis

### Local Contract Findings:

Contract: CrowdFundHYPCTPoolV3.Sol

---

ID B.1:

Status: **Resolved**

#### Low | Missing Validations in View Function

Present at: 'getProposalAssignmentNumber'@ L928

Description: The 'getProposalAssignmentNumber' function lacks validations for the 'proposalIndex' and 'tokenId' inputs. This can result in errors that are not clearly described.

Recommendation: Use the 'validIndex' modifier to check the validity of the 'proposalIndex' and add a 'require' statement to check the validity of the 'tokenId'.

Response: Fixed in 9fc4f46780540d0f7997c3f2e23944e1739b7216

---

ID B.2:

Status: **Resolved**

#### Low | Missing Validation on Constructor Parameter

Present at: 'constructor'@ L353

Description: The 'defaultFee' parameter lacks a validation check and does not have a maximum allowed value.

Recommendation: Consider adding a 'require' statement to check the upper bound of the possible pool fee. If the fee is paid in HyPC tokens, the maximum amount should probably be the total supply or a lower value.

Response: Fixed in 9fc4f46780540d0f7997c3f2e23944e1739b7216

## Complete Analysis

ID B.3:

Status: **Resolved**

**Informational | Improve Validation Logic**

Present at: 'transferDeposit', 'transferProposal' @L492, L520

Description: The 'transferDeposit' and 'transferProposal' functions uses an if-else-if validation structure at the beginning, which can be simplified for better readability and maintainability.

Recommendation: Simplify the validation logic by using separate require statements for each condition or use separate 'if' statements. This will improve code clarity and make the validation checks more straightforward.

Response: Fixed in 9fc4f46780540d0f7997c3f2e23944e1739b7216

ID B.4:

Status: **Resolved**

**Low | Duplicate Code**

Present at: 'redeemTokens', 'redeemSingleToken' @L601, L638

Description: The 'redeemTokens' and 'redeemSingleToken' functions contain identical logic for redeeming tokens. This redundancy can be refactored to improve code maintainability.

Recommendation: Consider adding an internal function '\_redeemToken' that encapsulates the shared token redemption logic. This internal function can then be called by both 'redeemTokens' and 'redeemSingleToken'.

Response: Fixed in 9fc4f46780540d0f7997c3f2e23944e1739b7216

## Complete Analysis

ID B.5:

Status: **Resolved**

### Low | Misleading Revert Message Description

Present at: 'redeemTokens' @L608

Description: In the 'redeemTokens' function, there is a validation to ensure the owner is the msg.sender. However, the error message describes it as 'ProposalMustBeCompleted'.

Recommendation: Consider separating this validation And use your already existing modifier "proposalOwner".

Response: Fixed in 9fc4f46780540d0f7997c3f2e23944e1739b7216 and uses ProposalMustBeCompletedOrOwner

ID B.6:

Status: **Resolved**

### Low | Missing Sanity Check for 'level' Parameter

Present at: 'swapTokens' @L567

Description: In the swapTokens function, there is no validation for the 'level' input parameter. This lack of validation can allow invalid or unexpected values for level, which may lead to unintended behavior or vulnerabilities in the contract.

Recommendation: Consider adding a 'require' statement to ensure that the level parameter is within the expected range. For example, require(level >= 10 && level <= 19, "Invalid level"); will enforce that level is between 10 and 19, inclusive.

Response: Fixed in 9fc4f46780540d0f7997c3f2e23944e1739b7216

## Complete Analysis

ID B.7

Status: Resolved

### Low | Duplicate Calculation

Present at: 'swapTokens' @L577

Description: In the 'swapTokens' function, the calculation '2\*\*(level-10)' is performed multiple times. This redundancy increases gas consumption unnecessarily, as the calculation yields the same result each time.

Recommendation: Store the result of 2\*\*(level-10) in a variable and reuse it throughout the function to optimize gas usage. For example, uint256 swapMultiplier = 2\*\*(level - 10); can be used to hold the value and replace subsequent occurrences of the calculation.

Response: Fixed in 9fc4f46780540d0f7997c3f2e23944e1739b7216

ID B.8:

Status: Acknowledged

### Medium | Wrong 'nftsSwapped' Calculation Logic

Present at: 'swapTokens' @L578

Description: The calculation for 'nftsSwapped' in the swapTokens function appears to be incorrect. According to the current logic, 'nftsSwapped' should simply equal tokensToSwap as that's the number of tokens that are actually swapped.

Recommendation: Review and possibly adjust the calculation logic to ensure it aligns with the intended behavior. The 'nftsSwapped' probably need to be equal only to 'tokensToSwap'

Response: nftsSwapped refers to the number of equivalent level 10 tokens inside the proposal. Since PoolV3 supports SwapV2, the tokens inside a proposal can be at different levels. As such, the nftsSwapped and numberNFTs values indicate the number of equivalent level 10 nfts that

## Complete Analysis

---

have been swapped inside the proposal and the total number that be swapped inside the proposal respectively.

For example, a proposal with numberNFTs = 9, can have nine level 10 cHyPcs inside the tokenIds array, or seven level 10s and one level 11, or one level 10 and one level 13.

In the case of tokensToSwap, this argument will swap the number of actual tokens inside the tokenIds array, while modifying the nftsSwapped value according to the level of each of those tokens being removed from the array.

---

ID B.9:

Status: Acknowledged

### Discussion | Clarification on Token Redemption Before Proposal Completion

Present at: 'redeemTokens' @L607

Description: The 'redeemTokens' function allows the owner of a proposal to redeem tokens even before the proposal has been completed. The function contains a condition that permits the owner (msg.sender) to redeem tokens if the proposal status is not 'Term.COMPLETED'. This behavior raises questions regarding the intended use case and security implications.

Recommendation: Provide clarification on the rationale behind permitting the owner to redeem tokens before the proposal's completion.

Response: There will only be tokens to redeem inside the contract if the proposal Term is either Term.STARTED or Term.COMPLETED. Like in the PoolV2, if the proposal is in the STARTED state, then the proposal owner is allowed to swap and redeem tokens for the proposal however it makes sense for them to do token assignments. However, once the proposal is in the Term.COMPLETED state, then tokens can no longer be swapped for, but anyone can redeem tokens for the proposal since depositors will likely need to ask the contract to redeem the tokens for HyPC so they can then withdraw their HyPC from the contract.

## Complete Analysis

---

ID B.10

Status: Acknowledged

### Medium | Incorrect 'nftsSwapped' Calculation Logic

Present at: 'redeemToken', 'redeemSingleToken' @L619, L651

Description: As described in issue B.8, this issue concerns the same subject. In both redeem functions, after redemption, the value of nftsSwapped is updated and reduced by 'tokensToRedeem \* 2\*\*(tokenLevel - 10)', which is incorrect, as only tokensToRedeem number of tokens are actually redeemed.

Recommendation: Consider removing the multiplier in the token level, consistent with the fix for issue B.8.

Response: See response to B.8. Also note that the tokenLevel can be different for each tokenId inside the tokenIds array. So nftsSwapped will not be reduced by 'tokensToRedeem \* 2\*\*(tokenLevel-10)' in general.

---

ID B.11:

Status: Resolved

### Discussion | APR computed only based on Level 10

Present at: 'createProposal' @L420

Description: on purchasing chypcs

Recommendation: Consider removing the multiplier in the token level based on the way that Isse B.8 was fixed

Response: See response B.8. Since cHyPC can be swapped at different levels in SwapV2, the numberNFTs value is based on the level 10 equivalent amount of NFTs instead, so the requiredFunds would be 'numberNFTs \* 1024 \* 1,000,000' as listed.

## Complete Analysis

ID B.12

Status: **Resolved**

### Low | Poor Naming Convention

Present at: 'redeemToken', 'redeemSingleToken' @L603, L640

Description: Both functions have a parameter named 'TokenToRedeem'. In the 'redeemToken' function, this parameter represents the amount of tokens to redeem, while in the 'redeemSingleToken' function, it represents the token index that the user wishes to redeem.

Recommendation: Consider renaming the parameter in the 'redeemSingleToken' function to 'TokenIndexToRedeem' to improve clarity and avoid confusion.

Response: Fixed in 9fc4f46780540d0f7997c3f2e23944e1739b7216

ID B.13:

Status: **Resolved**

### High | Incorrect Token Removal in redeemSingleToken Function

Present at: 'redeemToken', 'redeemSingleToken' @L603, L640

Description: The redeemSingleToken function allows a user to specify a token index to redeem. However, the function contains a line 'proposals[proposalIndex].tokenIds.pop();' which removes the last element in the tokenIds array instead of the specified token index. This could lead to incorrect token removal and potential loss of the intended token.

Recommendation: The corrected logic should swap the specified token with the last element and then remove the last element to maintain array integrity.

Response: The line directly above the pop call copies the last element of the list to the tokenIndex position before popping it, therefore removing the tokenIndex element from the list.

## Complete Analysis

ID B.14

Status: Resolved

### High | Use of Memory Instead of Storage

Present at: 'redeemToken', 'redeemSingleToken' @L603, L640

Description: The 'redeemSingleToken' function and the 'redeemToken' function declares ContractProposal as a memory variable, which means any modifications to it will not affect the actual data stored in the contract. This could lead to issues where changes made within the function do not persist, causing incorrect contract behavior.

Recommendation: Modify the function to declare ContractProposal as a storage variable to ensure that changes persist. This will allow the function to correctly modify the state of the contract.

Response: As from the previous audit, memory is used instead of storage to lower gas costs. When a change is intended to be persisted, proposals[proposalIndex] is used instead of proposalData.

ID B.15:

Status: Acknowledged

### Discussion | Apr Calculation Only For Level 10

Present at: 'createProposal' @L420

Description: It is unclear why the 'requiredFund' parameter, which is part of the APR calculation, is calculated only for level 10. The difference can be set into the 'numberNFTs' parameter. For example, if level 19 is required, then 'numberNFTs' should be ' $\langle \text{real\_number\_of\_NFTs} \rangle * 2^{**9}$ '. This also affects the calculation in the 'swapTokens' function, making the code unclear.

Recommendation: Consider adding a level parameter to 'createProposal'. Make sure this is an intentional logic.

## Complete Analysis

---

Response: As mentioned in B.8, since a started proposal can swap for any combination of token levels from 10-19, the numberNFTs is based off of the level 10 equivalents (level 10 being the lowest common denominator). So setting a proposal for 512 numberNFTs, means you can swap one level 19 token, or two level 18 tokens, or one level 18 token and two level 17 tokens, or any combination that will sum to the same amount.

**Disclaimer:**

DcentraLab Diligence (DD) has provided the code to the client as is and assumes no responsibility nor legal liability for any use client may do with the code. Any and all usage and/or deployment of the code provided by DcentraLab Diligence will be done solely by the client, at the sole discretion, responsibility, risk, and legal liability of the Client, and DD will not be held accountable or liable for any loss of funds, security exploits or incidents, or any other unintended or negative outcome that may occur in relation to the code provided by DD.

This report is not, nor should be considered, an "endorsement" or "disapproval" of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any "product" or "asset" created by any team or project that contracts DD to perform a security assessment. This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model, or legal compliance.

This report and the provided code or services as part of the SOW pertaining to this report should not be used in any way to make decisions around investment or involvement with any particular project. This report in no way provides investment advice, nor should it be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk. DD's position is that each company and individual are responsible for their own due diligence and continuous security. DD's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.

The assessment services provided by DD are subject to dependencies and are under continuing development. You agree that your access and/or use, including but not limited to any services, code, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives, false negatives, and other unpredictable results. The services may access, and depend upon, multiple layers of third parties.

ALL SERVICES, THE LABELS, THE ASSESSMENT REPORT, WORK PRODUCT, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AS AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, DcentraLab Diligence (DD) HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE SERVICES, ASSESSMENT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, DD SPECIFICALLY DISCLAIMS

ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM THE COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

WITHOUT LIMITING THE FOREGOING, DD MAKES NO WARRANTY OF ANY KIND THAT THE SERVICES, THE LABELS, THE ASSESSMENT / VERIFICATION REPORT, WORK PRODUCT, CODE OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET THE CUSTOMER'S OR ANY OTHER PERSON'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE.

WITHOUT LIMITATION TO THE DISCLAIMER HyperCycle Contracts FOREGOING, DD PROVIDES NO WARRANTY OR UNDERTAKING, AND MAKES NO REPRESENTATION OF ANY KIND THAT THE SERVICE WILL MEET THE CUSTOMER'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULTS, BE COMPATIBLE OR WORK WITH ANY OTHER SOFTWARE, APPLICATIONS, SYSTEMS OR SERVICES, OPERATE WITHOUT INTERRUPTION, MEET ANY PERFORMANCE OR RELIABILITY STANDARDS OR BE ERROR-FREE OR THAT ANY ERRORS OR DEFECTS CAN OR WILL BE CORRECTED.

WITHOUT LIMITING THE FOREGOING, NEITHER DD NOR ANY OF DD'S AGENTS MAKES ANY REPRESENTATION OR WARRANTY OF ANY KIND, EXPRESS OR IMPLIED AS TO THE ACCURACY, RELIABILITY, OR CURRENCY OF ANY INFORMATION, CODE OR CONTENT PROVIDED THROUGH THE SERVICE. DD WILL ASSUME NO LIABILITY OR RESPONSIBILITY FOR (I) ANY ERRORS, MISTAKES, OR INACCURACIES OF CONTENT AND MATERIALS OR FOR ANY LOSS OR DAMAGE OF ANY KIND INCURRED AS A RESULT OF THE USE OF ANY CONTENT OR CODE, OR (II) ANY PERSONAL INJURY OR PROPERTY DAMAGE, OF ANY NATURE WHATSOEVER, RESULTING FROM CUSTOMER'S ACCESS TO OR USE OF THE SERVICES, CODE, ASSESSMENT REPORT, OR OTHER MATERIALS.

ALL THIRD-PARTY MATERIALS ARE PROVIDED "AS IS," AND ANY REPRESENTATION OR WARRANTY OF OR CONCERNING ANY THIRD-PARTY MATERIALS IS STRICTLY BETWEEN THE CUSTOMER AND THE THIRD-PARTY OWNER OR DISTRIBUTOR OF THE THIRD-PARTY MATERIALS. THE SERVICES, ASSESSMENT REPORT, AND ANY OTHER MATERIALS HEREUNDER ARE SOLELY PROVIDED TO THE CUSTOMER AND MAY NOT BE RELIED ON BY ANY OTHER PERSON OR FOR ANY PURPOSE NOT SPECIFICALLY IDENTIFIED IN THIS AGREEMENT, NOR MAY COPIES BE DELIVERED TO, ANY OTHER PERSON WITHOUT DD'S PRIOR WRITTEN CONSENT IN EACH INSTANCE.

NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST DD WITH RESPECT TO SUCH SERVICES, ASSESSMENT REPORT, AND ANY ACCOMPANYING MATERIALS. THE REPRESENTATIONS AND WARRANTIES OF DD CONTAINED IN THIS AGREEMENT ARE SOLELY FOR THE BENEFIT OF THE CUSTOMER. ACCORDINGLY, NO THIRD PARTY OR ANYONE ACTING ON BEHALF OF ANY THEREOF, SHALL BE A THIRD PARTY OR OTHER BENEFICIARY OF SUCH REPRESENTATIONS AND WARRANTIES AND NO SUCH THIRD PARTY SHALL HAVE ANY RIGHTS OF CONTRIBUTION AGAINST DD WITH RESPECT TO SUCH REPRESENTATIONS OR WARRANTIES OR ANY MATTER SUBJECT TO OR RESULTING IN INDEMNIFICATION UNDER THIS AGREEMENT OR OTHERWISE. FOR AVOIDANCE OF DOUBT, THE SERVICES, INCLUDING ANY ASSOCIATED ASSESSMENT REPORTS, CODE, OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

[dcentralab.com/diligence](http://dcentralab.com/diligence)



# DcentralLab Diligence